



**VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ**

BRNO UNIVERSITY OF TECHNOLOGY

**FAKULTA STROJNÍHO INŽENÝRSTVÍ**

FACULTY OF MECHANICAL ENGINEERING

**ÚSTAV MECHANIKY TĚLES, MECHATRONIKY A  
BIOMECHANIKY**

INSTITUTE OF SOLID MECHANICS, MECHATRONICS AND BIOMECHANICS

**ZPRACOVÁNÍ OBRAZU A HLOUBKOVÉ MAPY PRO  
VYTVOŘENÍ INTERAKTIVNÍHO MODELU**

IMAGE PROCESSING AND DEPTH MAPS TO CREATE AN INTERACTIVE MODEL

**BAKALÁŘSKÁ PRÁCE**

BACHELOR'S THESIS

**AUTOR PRÁCE**

AUTHOR

**Lukáš Gelo**

**VEDOUCÍ PRÁCE**

SUPERVISOR

**Ing. Martin Appel**

**BRNO 2020**

# Zadání bakalářské práce

Ústav: Ústav mechaniky těles, mechatroniky a biomechaniky  
Student: **Lukáš Gelo**  
Studijní program: Aplikované vědy v inženýrství  
Studijní obor: Mechatronika  
Vedoucí práce: **Ing. Martin Appel**  
Akademický rok: 2019/20

Ředitel ústavu Vám v souladu se zákonem č.111/1998 o vysokých školách a se Studijním a zkušebním řádem VUT v Brně určuje následující téma bakalářské práce:

## **Zpracování obrazu a hloubkové mapy pro vytvoření interaktivního modelu**

### **Stručná charakteristika problematiky úkolu:**

Motivací práce bude ověřit možnosti dostupných nástrojů v programu MATLAB pro zpracování hloubkové mapy a obrazu. MATLAB umožňuje implementovat do programu rozpoznávání obličejů, sledování pohyblivých předmětů a další pokročilou práci s obrazem. Hloubková mapa umožňuje získat informace, které kamera nezjistí, jako je například prostorová geometrie předmětu. S takto získanou informací lze různě nakládat. Lze například nalézt v obraze určité geometrické tvary. Také rozpoznávání lidí a gest je robustnější než s využitím jen kamerových dat.

Výsledkem bude sada demonstračních ukázek, implementovaná do mechatronického exponátu. Tvorba mechatronického exponátu je cílem jiné práce a tato práce se jeho tvorbou nezabývá.

### **Cíle bakalářské práce:**

- 1) Provedte průzkum trhu v oblasti hloubkových snímačů.
- 2) Vytvořte program, který najde v obraze určité objekty s pomocí hloubkové mapy.
- 3) Vytvořte komunikační protokol pro vytvořený program, který umožní implementaci do rozsáhlejšího programu.
- 4) Implementujte program do mechatronického exponátu.

### **Seznam doporučené literatury:**

CORKE, Peter I. Robotics, vision and control: fundamental algorithms in MATLAB. Berlin: Springer, 2011. Springer tracts in advanced robotics, v. 73. ISBN 9783642201431.

VALÁŠEK, Michael. Mechatronika. Praha: České vysoké učení technické, 1995. ISBN 80-01-01276-X.

GREPL, Robert. Kinematika a dynamika mechatronických systémů. Brno: Akademické nakladatelství CERM, 2007. ISBN 978-80-214-3530-8.

Termín odevzdání bakalářské práce je stanoven časovým plánem akademického roku 2019/20

V Brně, dne

L. S.

---

prof. Ing. Jindřich Petruška, CSc.  
ředitel ústavu

---

doc. Ing. Jaroslav Katolický, Ph.D.  
děkan fakulty

# Abstrakt

Táto bakalárska práca sa zaoberá vytvorením komunikačného protokolu pre kamery Intel® RealSense™ v programe MATLAB, vytvorením programu, ktorý nájde v obraze určité predmety za použitia hĺbkovej mapy a implementáciou tohto programu do mechatronického exponátu.

Teoretická časť sa zaoberá metódami snímania vzdialenosti a hĺbky. Objasňuje jednotlivé princípy bezkontaktného nedeštruktívneho merania vzdialenosti a hĺbky spolu s porovnaním ich výhod a nevýhod.

# Abstract

This bachelor's thesis deals with creation of communication protocol for Intel® RealSense™ cameras in MATLAB, creation of program which finds certain objects in image using depth map and implementation of this program into mechatronic exponate.

Theoretical part deals with methods of depth and distance scanning. It clarifies principles of contactless and nondestructive measurement of depth and distance along with comparison of their advantages and disadvantages.

# Klíčové slová

hĺbková mapa, manipulátor, mračno bodov, RealSense, RGB-D, rozoznanie objektov

# Keywords

depth map, manipulator, pointcloud, RealSense, RGB-D, object recognition

# Bibliografická citácia

GELO, Lukáš. *Zpracování obrazu a hloubkové mapy pro vytvoření interaktivního modelu*. Brno, 2020. Dostupné také z: <https://www.vutbr.cz/studenti/zav-prace/detail/125194>. Bakalářská práce. Vysoké učení technické v Brně, Fakulta strojního inženýrství, Ústav mechaniky těles, mechatroniky a biomechaniky. Vedoucí práce Ing. Martin Appel.



Prehlasujem, že som túto bakalársku prácu vypracoval sám pod vedením Ing. Martina Appela a uviedol som všetky zdroje a publikácie, z ktorých som čerpal.

**Lukáš Gelo**

Brno . . . . .

. . . . .

Týmto by som rád poďakoval vedúcemu svojej bakalárskej práce Ing. Martinovi Appelovi za jeho pomoc a cenné rady pri vypracovaní tejto práce a Bc. Michalovi Mudrákovi za poskytnutie ochoty a poskytnutie prostriedkov pre výrobu dielov exponátu.

**Lukáš Gelo**

# Obsah

<b>Úvod</b>	<b>9</b>
<b>1 Metódy bezkontaktného merania hĺbky</b>	<b>10</b>
1.1 Meranie doby letu . . . . .	10
1.1.1 Microsoft® Kinect™ 2.0 . . . . .	14
1.1.2 Senzor IFM O3D303 . . . . .	15
1.2 Štruktúrované svetlo . . . . .	16
1.2.1 Microsoft® Kinect™ 1.0 . . . . .	18
1.2.2 Asus® XtrionPro™ Live . . . . .	18
1.2.3 Zivid One <sup>+</sup> . . . . .	19
1.2.4 PhoXi 3D skener . . . . .	21
1.3 Aktívne a pasívne stereo snímanie . . . . .	22
1.3.1 MultiSense S7 . . . . .	24
1.3.2 ZED . . . . .	25
1.3.3 Intel® RealSense™ . . . . .	26
1.4 Zhrnutie a porovnanie metód merania hĺbky . . . . .	27
<b>2 Intel® RealSense™ rady D400</b>	<b>29</b>
<b>3 Komunikačný protokol</b>	<b>30</b>
3.1 Podporný balík pre prostredie MATLAB . . . . .	30
3.2 Objekt RealSense . . . . .	31
3.2.1 Inicializácia objektu RealSense . . . . .	32
3.2.2 Nastavenie a spustenie dátového toku . . . . .	34
3.2.3 Nastavenie zariadenia . . . . .	37
3.2.4 Získavanie a postprocesovanie dát . . . . .	39

3.3	Testovanie rýchlosti snímania a presnosti merania . . . . .	45
<b>4</b>	<b>Spracovanie dát zo zariadenia RealSense</b>	<b>51</b>
<b>5</b>	<b>Návrh mechatronického exponátu</b>	<b>55</b>
5.1	Manipulátor verzia 1.0 . . . . .	55
5.2	Manipulátor verzia 2.0 . . . . .	57
5.3	Výroba manipulátoru . . . . .	60
5.4	Elektrické zapojenie . . . . .	61
5.5	Výpočet uhlov natočenia kĺbov a servomotorov . . . . .	62
5.6	Software a simulácia . . . . .	64
5.7	Implementácia snímania scény a spracovania obrazu do exponátu . . . . .	67
	<b>Záver</b>	<b>70</b>
	<b>Literatúra</b>	<b>72</b>
	<b>Zoznam skratiek</b>	<b>75</b>
	<b>Zoznam obrázkov</b>	<b>76</b>
	<b>Zoznam tabuliek</b>	<b>79</b>
	<b>Prílohy</b>	<b>80</b>

# Úvod

Zrak je najdôležitejším ľudským zmyslom v rámci orientácie v prostredí. Spracovanie obrazu je dôležitým bodom v oblasti automatizácie, autonómnych zariadení a zabezpečenia. Existujú však aplikácie, pri ktorých získavanie 2D obrazu nie je dostačujúce. V prípadoch, kedy je potrebné zistiť vzdialenosť, tvar alebo rozmer objektov či štruktúru okolia je nutné použiť zariadenia a senzory merajúce hĺbku. V tejto práci je popísané využitie jedného z týchto zariadení, a to senzoru Intel® RealSense™ D415. Časť práce sa zaoberá možnosťou využitia senzoru v prostredí MATLAB a vytvorením komunikačného protokolu pre prácu so sensorom.

Ďalšou časťou práce je vytvorenie programu, ktorý v obraze nájde určité objekty za použitia algoritmov pre spracovanie hĺbkovej mapy. Hľadanie objektov v hĺbkovej mape má voči spracovaniu 2D obrazu výhody, ktoré sú v tejto práci taktiež popísané.

Využitie vyššie zmieneného programu je predstavené na mechatronickom exponáte, ktorého návrh je taktiež súčasťou tejto práce.

V nasledujúcej kapitole sú predstavené metódy snímania hĺbky spolu s príkladmi hĺbkových senzorov dostupných na dnešnom trhu.

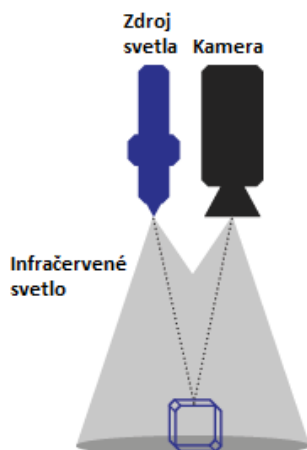
# 1 Metódy bezkontaktného merania hĺbky

Hĺbkou je možné merať senzormi alebo zariadeniami využívajúcimi metódy kontaktného merania, ako je koordinačný merací prístroj (CMM), a deštruktívnymi metódami odstraňovaním jednotlivých vrstiev telesa alebo bezkontaktné, využitím jednej z reflexných metód. Táto práca sa zaoberá práve princípom a využitím reflexných metód snímania hĺbky.

Spracovanie obrazu sa stalo súčasťou všedného života. Je možné sa s ním stretnúť napríklad pri zabezpečení objektov a majetku (odomykanie mobilných zariadení tvárou, bezpečnostné systémy) alebo pri sledovaní akosti výroby. Tu všade je možné nájsť zariadenia snímajúce a spracovávajúce dvojrozmerný obraz. Pridaním možnosti snímania tretieho rozmeru, hĺbky prostredia, je možné tieto algoritmy zrýchliť, napr. rozdelením obrazu na objekty záujmu a pozadie, a rozšíriť o nové funkcie ako je rozoznávanie gest, biometria tváre a pod. Snímanie trojrozmerného obrazu je všeobecne náročnejšie na výpočetný čas a komplexnosť softwarového riešenia. V nasledujúcich kapitolách sú predstavené rôzne metódy snímania hĺbky spolu so senzormi a zariadeniami využívajúcimi tieto metódy dostupnými na dnešnom trhu.

## 1.1 Meranie doby letu

Metóda merania doby letu (z angl. Time of Flight - TOF) spočíva v meraní fázového posuvu medzi vysielaným svetelným lúčom a odrazom, teda meria priamo vzdialenosť prostredníctvom fázového posuvu signálu. Základnými časťami zariadení merajúcich vzdialenosť využitím metódy merania doby letu sú zdroj osvetlenia, zvyčajne laser alebo LED dióda, vysielajúci lúče z rozmedzia pásiem infračerveného (IR) a viditeľného svetla a prijímač, zobrazovací senzor, ktorý premieňa energiu zachytených fotónov na elektrickú energiu. Zobrazovací senzor zaznamenáva vlnenie o rovnakej vlnovej dĺžke, aké je generované vysielateľom. Prijímač je zostavený z viacerých senzorov tvoriacich rad pixelov zaznamenávajúcich vzdialenosť viacerých bodov súčasne. [1][2][3]

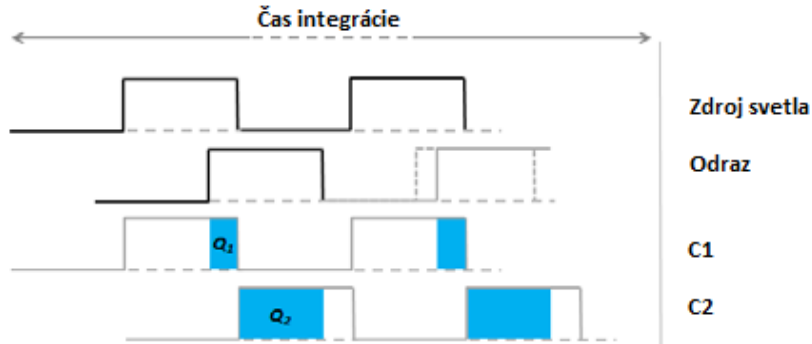


Obr. 1.1: Meranie vzdialenosti použitím metódy TOF [2]

## 1 METÓDY BEZKONTAKTNÉHO MERANIA HĽBKY

Podľa zvolenej metódy výpočtu vzdialenosti je potrebný väčší počet takzvaných subsnímok jedného merania, z ktorých sa určí fázový posun, energia dopadnutých fotónov a následne vzdialenosť. Každý pixel je rozdelený podľa počtu subsnímok na daný počet dielov, ktoré sa rýchlo „otvárajú a zatvárajú“. V stave otvorenom pixel prijíma fotóny, v stave zatvorenom ich neprijíma. [1][2][3]

Prvý zo spôsobov určenia vzdialenosti je metóda pulzného zdroja svetla, kedy zdroj svetla vysiela svetelné impulzy. Tvar svetelných impulzov môže byť sínusový, ale častejšie sa využíva generovanie obdĺžnikového tvaru signálu, ktorý je jednoducho generovateľný digitálnym obvodom. [1][2]



Obr. 1.2: Princíp merania vzdialenosti použitím metódy TOF s pulzným zdrojom svetla [1]

Zdroj svetla svieti po krátku dobu  $\Delta t$ , sú zaznamenané dva subsnímky merania *C1* a *C2*, ako je možné vidieť na obrázku 1.2. To znamená, že je pixel rozdelený na dva subpixely, ich otvorenie a zatvorenie je navzájom fázovo posunuté o  $180^\circ$ , pričom otváranie jedného zo subpixelov je v rovnakej fáze so zdrojom svetla. V každom subpixeli je určená energia fotónov  $Q1$  a  $Q2$  a následne sa použitím vzťahu 1.1, kde  $c$  je rýchlosť svetla, určí vzdialenosť  $d$ . [1]

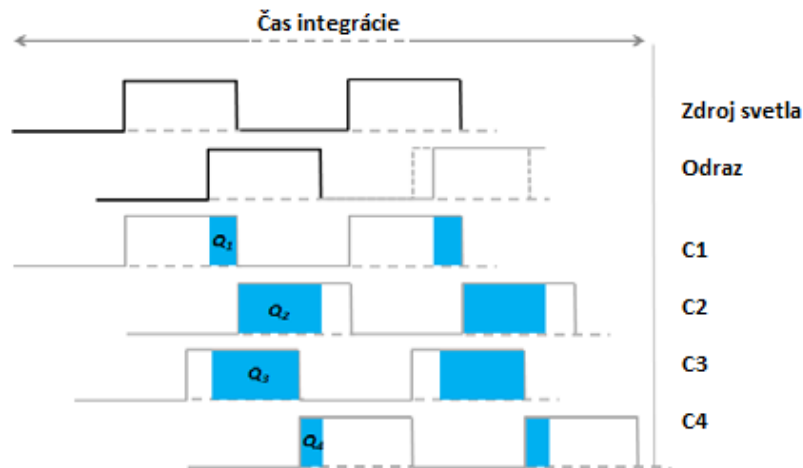
$$d = \frac{1}{2} \cdot c \cdot \Delta t \cdot \left( \frac{Q_2}{Q_1 + Q_2} \right) \quad (1.1)$$

Nevýhodou tejto metódy je vznik chyby zapríčinennej okolitým svetlom. Táto chyba merania je odstránená v druhej metóde určovania vzdialenosti, metóde kontinuálnej vlny (CW), kedy zdroj svetla žiari neustále generujúc periodické svetelné impulzy. [1]

K určeniu vzdialenosti sú potrebné štyri subsnímky, čo znamená, že sú pixely rozdelené na štyri časti so vzájomným fázovým posunom otvárania subpixelu o  $90$  stupňov. Otváranie jedného zo subpixelov je vo fázi s generovanými svetelnými impulzmi. Vzdialenosť je možné určiť použitím vzťahu 1.2, kde  $c$  je rýchlosť svetla,  $f$  je frekvencia vysielaných svetelných impulzov a  $Q_1$  až  $Q_4$  sú energie dopadnutých fotónov. [1]

$$d = \frac{c}{4 \cdot \pi \cdot f} \cdot \arctan \left( \frac{Q_3 - Q_4}{Q_1 - Q_2} \right) \quad (1.2)$$

Použitím rozdielu hodnôt medzi dvoma subsnímami ( $Q_1 - Q_2$ ) a ( $Q_3 - Q_4$ ) je odstránená systematická chyba merania zapríčinená okolitým osvetlením. Táto metóda má však taktiež nevýhody. Jednou z nich je obmedzenie maximálnej merateľnej vzdialenosti. Nameraný fázový



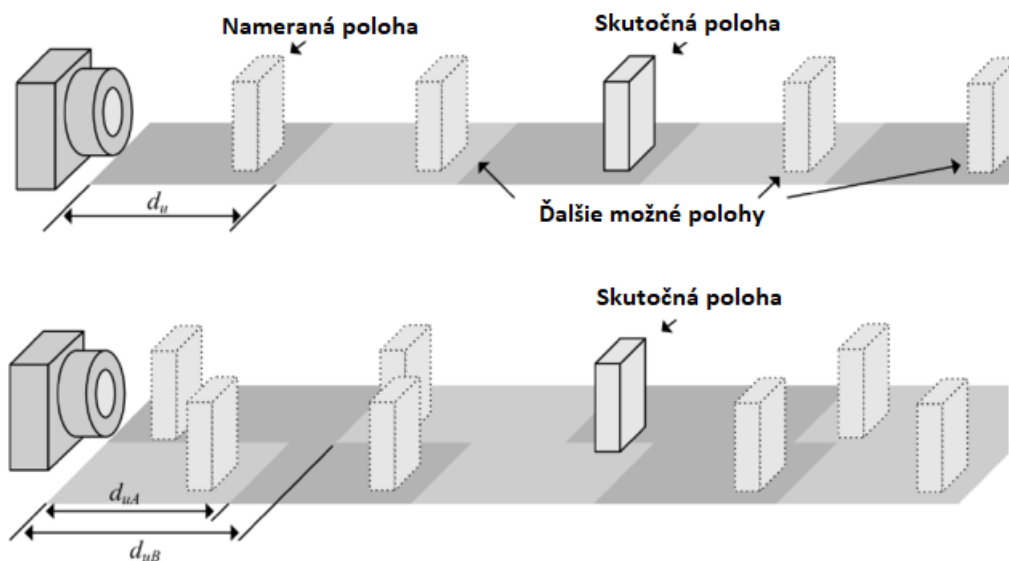
Obr. 1.3: Princíp merania vzdialenosti použitím metódy TOF CW [1]

posun odpovedá vzdialenosti  $d$ , ale v dôsledku fázovej nejednoznačnosti taktiež vzdialenosti  $2d$ ,  $3d$ , atď., čo obmedzuje maximálnu merateľnú vzdialenosť na vzdialenosť určenú vzťahom 1.3, kde  $c$  je rýchlosť svetla a  $f$  frekvencia vysielaných svetelných impulzov.

$$d_{max} = \frac{c}{2 \cdot f} \quad (1.3)$$

Keďže rýchlosť svetla v danom prostredí je nemenná, je možné maximálnu merateľnú vzdialenosť ovplyvniť zmenšením frekvencie vysielaných impulzov, čo má však za následok zväčšenie odchýlky merania. [1][3]

Pre zväčšenie maximálnej merateľnej vzdialenosti sa využíva tretia metóda a to metóda modulovania multifrekvenčných svetelných pulzov. Princíp spočíva v použití viacerých modulovaných frekvencií vysielaného paprsku počas jedného merania. Výsledkom výpočtu vzdialenosti pre každú z použitých frekvencií je množina hodnôt vzdialenosti. Prienik týchto množín, teda hodnota vzdialenosti vyskytujúca sa vo všetkých množinách, je správna hodnota vzdialenosti. Princíp merania vzdialenosti multifrekvenčnou metódou je zobrazený na obrázku 1.4. [1]



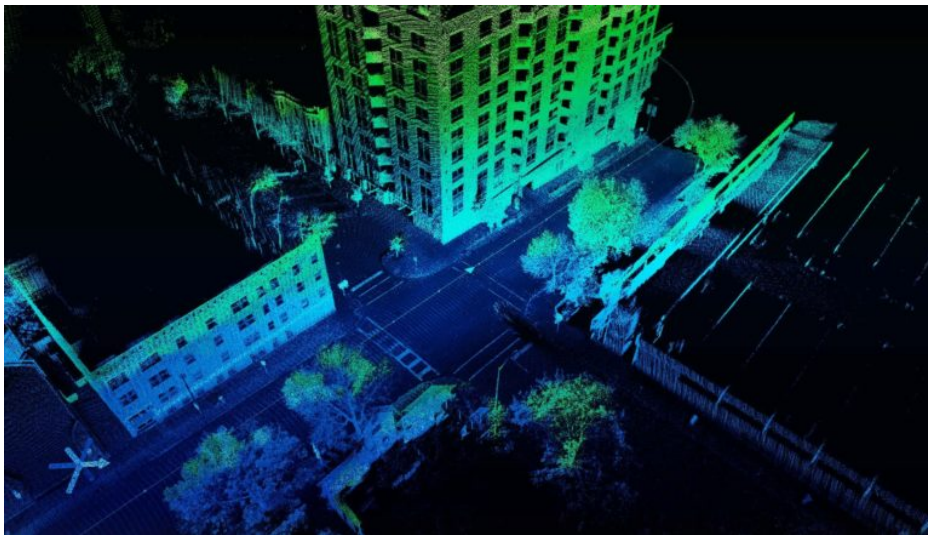
Obr. 1.4: Princíp merania vzdialenosti použitím multifrekvenčnej metódy TOF [1]



## 1 METÓDY BEZKONTAKTNÉHO MERANIA HĽBKY

Výhodou metódy TOF je veľké rozmedzie merateľnej vzdialenosti, kedy je možné touto metódou merať na malú, ale aj na veľmi veľkú vzdialenosť. Meranie metódou TOF je možné použiť na priamom slnečnom svetle a taktiež je ňou možné dosiahnuť veľkú rýchlosť snímokovania. Farba a drsnosť povrchu snímaných objektov nemá na presnosť merania vplyv. Avšak metóda TOF má taktiež svoje nevýhody v porovnaní s ostatnými metódami. Jednou z nich je pomerne nízke rozlíšenie a tým aj presnosť v rovine XY. Taktiež poloha senzoru voči okolitému šasi zariadenia je dôležitá - odrážanie lúčov vysielaného svetla od šasi často spôsobuje systematickú chybu merania. Ďalším rizikom je efekt viacestného odrazu, kedy sa odrazený paprsok môže viackrát odraziť než dorazí k snímaču. [1][2][3]

Na princípe merania doby letu funguje taktiež technológia LIDaR. Jedná sa o technológiu, pri ktorej je ako zdroj svetla využívaný laser emitujúci svetelné paprsky zväčša v infračervenom (zriedka viditeľnom alebo ultrafialovom) spektre. Laser môže vyžarovať súčasne paprsky o rôznych vlnových dĺžkach za účelom snímania rôznych objektov a materiálov, ako je napr. voda, vegetácia, pôda atď. Laser je umiestnený na rotujúcej hlavici, ktorá umožňuje snímať priestor s  $360^\circ$  zornom uhlom, pričom získané dáta sú vo valcových alebo sférických súradniciach. V závislosti na výkone laseru je možné merať vzdialenosti v rozmedzí centimetrov až desiatok kilometrov. [4]



Obr. 1.5: Snímka vytvorená LIDaR zariadením [5]

Technológia LIDaR je využívaná napr. pri topografii, mapovaní exteriéru a interiéru budov alebo aj v doprave pre snímání okolia automobilu.

## 1.1.1 Microsoft® Kinect™ 2.0

Kamera Microsoft Kinect 2.0, známa aj pod názvom Kinect v2, je hĺbková kamera od spoločnosti Microsoft merajúca hĺbku využitím metódy TOF. Primárnym využitím senzoru je ovládanie herných konzol, kde zastáva funkciu ovládača a umožňuje užívateľovi ovládať konzolu pohybom tela. Je schopný sledovať až šesť postáv a rozpoznať 25 kľbov. Zároveň je dostatočne presný na to, aby bol použitý na rozpoznávanie tváří. Jedná sa teda o všestranné zariadenie slúžiace nielen k snímaniu hĺbky, ale aj k sledovaniu postáv a rozoznávaniu pohybu a gest. [6]



Obr. 1.6: Kamera Microsoft® Kinect™ 2.0 [7]

Tabuľka 1.1: Základné parametre senzoru Microsoft® Kinect™ 2.0 [6]

Rozlíšenie farebnej kamery [px]	1920 x 1080
Rozlíšenie hĺbkového senzoru [px]	512 x 424
Merateľná hĺbka [m]	0.4 až 4.5
Rýchlosť snímkovania [S/s]	30
Zorné pole (H x V)	70° x 60°
Rozmery (V x Š x H) [mm]	45 x 250 x 70
Rozhranie	USB 3.0
Oneskorenie	min. 20 ms

Pre prácu so senzorom existujú knižnice v mnohých programovacích jazykoch ako napr. Python, C++, C#, Matlab a pod.

## 1.1.2 Senzor IFM O3D303

Senzor od spoločnosti IFM meria vzdialenosť využívajúc metódu TOF. Je vhodný pre priemyselné účely a použitie. Napájacie napätie senzoru je 20.4 až 28.8 Vdc pri spotrebe 10 W. Jeho vhodnosť pre priemyselné použitie podtrhuje výrobcom uvedený stupeň krytia IP65 a IP67 a ochrana elektrických obvodov voči skratu a prepätiu. Komunikácia je zabezpečená použitím ethernet rozhrania a komunikačným protokolom TCP/IP.[8]



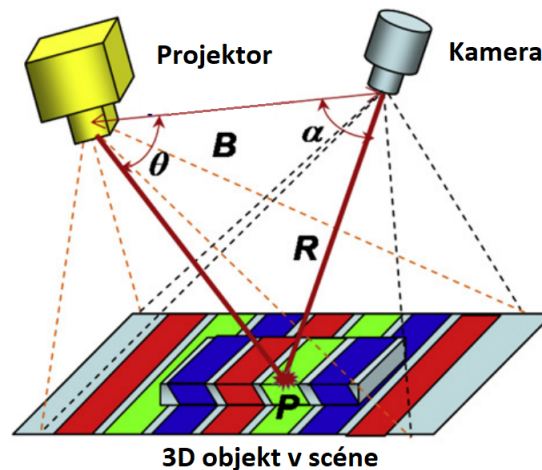
Obr. 1.7: Senzor IFM O3D303 [8]

Tabuľka 1.2: Základné parametre senzoru IFM O3D303 [8]

Rozlíšenie hĺbkového senzoru [px]	352 x 264
Merateľná hĺbka [m]	0.3 až 8
Rýchlosť snímkovania [S/s]	25
Zorné pole (H x V)	60° x 45°
Rozmery (V x Š x H) [mm]	72 x 65 x 82.6
Rozhranie	Ethernet
Oneskorenie	1 snímka

## 1.2 Štruktúrované svetlo

Metóda štruktúrovaného svetla (z angl. Structured Light - SL) k meraniu hĺbky využíva projekciu predom definovaného statického alebo meniaceho sa vzoru a sníma jeho deformáciu pomocou kamery. Svetlo používané pre projekciu je z oblasti viditeľného až infračerveného spektra. Hĺbka je meraná nepriamo využitím algoritmov triangulácie. Z dôvodu použitia triangulácie je potrebné poznať vzájomnú vzdialenosť projektoru a kamery - základňu (baseline). [3][9]



Obr. 1.8: Princíp triangulácie - metóda SL [9]

Vzdialenosť každého bodu (pixelu) môže byť určená použitím princípu triangulácie vyjadrujúc vzťah medzi polohou kamery, projektoru a snímaného bodu a je vyjadrená vzťahom 1.4. [9]

$$R = B \cdot \left( \frac{\sin \Phi}{\sin \alpha + \Phi} \right) \quad (1.4)$$

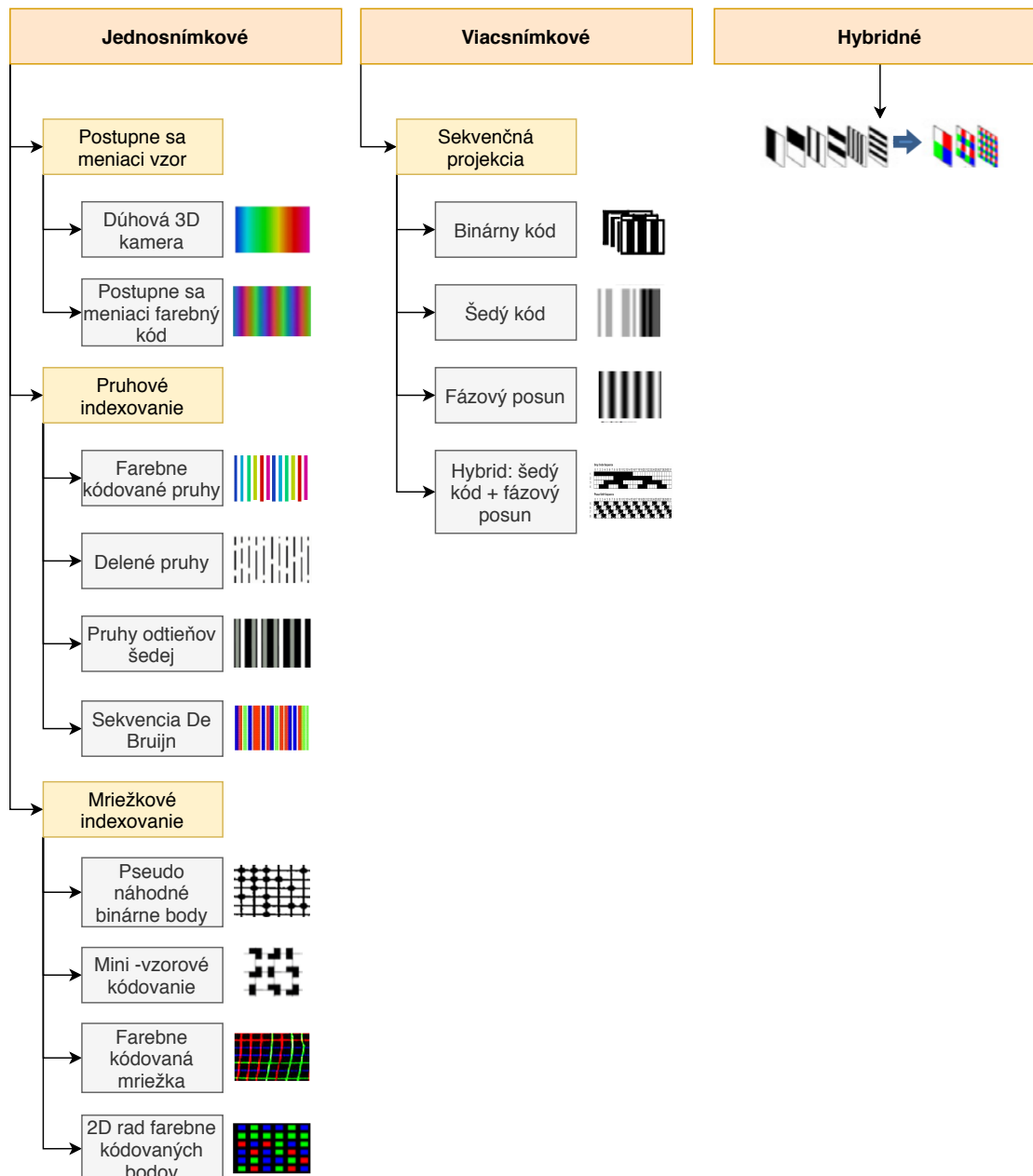
Ako je možné vidieť z obrázku 1.8,  $B$  je vzdialenosť medzi projektorom a kamerou,  $\alpha$  je uhol medzi základňou a priamkou spájajúcou snímaný bod  $P$  a kameru a  $\Phi$  je uhol medzi základňou a priamkou spájajúcou snímaný bod s projektorom. Projektor osvetľuje celú scénu snímanú kamerou jedným alebo sériou viacerých vzorov, pričom každý pixel snímanej scény je vo výsledku označený unikátnym vzorom alebo kódom, vďaka ktorému dokáže vyhodnocovací software identifikovať totožnosť bodu z hľadiska projektoru a kamery a metódou triangulácie vyhodnotiť jeho vzdialenosť. [2][9]

Ako zdroj svetla sa zväčša používa laser alebo LED dióda s optikou pre rozptýlenie lúčov. Z princípu funkcie metódy merania hĺbky pomocou projekcie rôznych vzorov na meranú scénu vyplýva, že je možné použiť ju aj v úplnej tme, keďže potrebné osvetlenie je vytvárané priamo senzorom. [2][3][9]

Metóda je vhodnejšia pre použitie v krytom slabšie osvetlenom prostredí, keďže presnosť vyhodnocovania vzorov a teda merania hĺbky sa zhoršuje s rastúcou intenzitou okolitého osvetlenia. Taktiež v dôsledku dlhšieho času potrebného pre spracovanie snímky je táto metóda vhodná najmä na snímanie hĺbky statickej alebo pomaly sa meniacej scény. [1][2][3]

Výhodou tohto spôsobu merania hĺbky je veľké rozlíšenie a presnosť v rovine XY pri použití DLP projektorov a kamier s vysokým rozlíšením.

# 1 METÓDY BEZKONTAKTNÉHO MERANIA HĽBKY



Obr. 1.9: Druhy vzorov pre projekciu [10]

Ako je možné vidieť v diagrame 1.9, existuje viac techník (rôznych vzorov) používaných pre projekciu. Podľa počtu vzorov, ktoré je potrebné premietnuť pre zhotovenie jedného merania, sa techniky delia do troch hlavných skupín a to na:

- jednosnímkové,
- multisnímkové,
- hybridné. [10]

Sekvenčná projekcia:

Použitím techniky sekvenčnej projekcie je možné dosiahnuť vysoké rozlíšenie a veľkú presnosť, avšak za cenu malej rýchlosti snímania v dôsledku nutnosti použitia veľkého množstva vzorov pre získanie jedného merania. Techniky sekvenčnej projekcie sú jednoducho implementovateľné, avšak sú vhodné pre snímání statických scén.

## 1 METÓDY BEZKONTAKTNÉHO MERANIA HĽBKY

Postupne sa meniaci vzor:

Techniky premietajúce postupne sa meniaci vzor dosahujú vysoké rozlíšenie merania, avšak sú veľmi citlivé na šum vznikajúci napríklad projekciou na lesklý povrch. Podobne ako pri sekvenčnej projekcii je vhodné používať ich na statické scény.

Pruhové a mriežkové indexovanie:

Voči vyššie spomenutým technikám majú nižšie rozlíšenie. Ich výhodou je používanie unikátnych vzorov a schopnosť snímať hýbajúce sa objekty bez vzniku rozsiahlej nepresnosti zapríčinené zmenou scény počas merania. [9]

Viac informácií o jednotlivých technikách projekcie je možné nájsť v dokumente [10].

### 1.2.1 Microsoft® Kinect™ 1.0

Kinect 1.0, známa aj pod názvom Kinect v1, je hĺbková kamera od spoločnosti Microsoft merajúca hĺbku využitím metódy SL. Ako je spomenuté v kapitole 1.2, princípom merania vzdialenosti pri využití metódy SL je projekcia známeho vzoru na snímanú scénu. Kinect v1 používa k projekcii techniku pseudo náhodných binárnych bodov, ktoré premieta v infračervenom spektre. IR svetlo je použité zámerne, keďže prvotným účelom senzoru bolo snímať postavu a pohyb užívateľa hrajúceho hru na hernej konzole. Svetlo neoslňuje a taktiež nie je zdraviu škodlivé, takže užívateľa počas hrania senzor nijako neohrozuje a nevyrušuje. Taktiež je použitím IR spektra obmedzené skreslenie pôsobením okolitého svetla. Kúpa nového senzoru už nie je možná, nakoľko bol nahradený vyššie spomínanou verziou 2.0. [6]



Obr. 1.10: Kamera Microsoft® Kinect™ 1.0 (vľavo) a premietaný vzor (vpravo)[6]

Výhodou použitia senzoru je jeho vstavaná funkcia rozoznávania a sledovania postáv a určenie kĺbov ľudského tela. Kinect v1 je schopný sledovať súčasne dve postavy a rozpoznať 20 kĺbov. [6]

Pre umožnenie práce so senzorom sú voľne dostupné knižnice v mnohých programovacích jazykoch ako napr. Python, C++, C#, Matlab a pod.

### 1.2.2 Asus® XtrionPro™ Live

Senzor od spoločnosti Asus®, podobne ako Microsoft® Kinect™ 1.0, meria vzdialenosť využitím metódy SL. Sníma zároveň hĺbkovú mapu, farebný obraz a zvuk, ktorý je snímaný použitím dvoch mikrofónov. [12]

Senzor je svojimi funkciami primárne určený na snímanie pohybu, polohy tela a gest užívateľa.

## 1 METÓDY BEZKONTAKTNÉHO MERANIA HĽBKY

Tabuľka 1.3: Základné parametre senzoru Microsoft® Kinect™ 1.0 [6]

Rozlíšenie farebnej kamery [px]	640 x 480
Rozlíšenie hĺbkového senzoru [px]	320 x 240
Merateľná hĺbka [m]	0.4 až 4
Rýchlosť snímkovania [S/s]	30
Zorné pole (H x V)	57° x 43°
Rozmery (V x Š x H) [mm]	180 x 355 x 76



Obr. 1.11: Senzor Asus® XtrionPro™ Live [11]

Asus XtrionPro Live je určený najmä pre vývojárov. Pre uľahčenie práce so senzorom sú voľne prístupné knižnice pre rôzne programovacie jazyky obsahujúce vstavané funkcie pre detekciu a sledovanie ľudského tela, rozpoznávanie pohybu a gest užívateľa. [12]

Tabuľka 1.4: Základné parametre senzoru Asus® XtrionPro™ Live [7]

Rozlíšenie farebnej kamery [px]	1280 x 1024
Rozlíšenie hĺbkového senzoru [px]	640 x 480
Merateľná hĺbka [m]	0.8 až 3.5
Rýchlosť snímkovania [S/s]	30
Zorné pole (H x V)	58° x 45°
Rozmery (V x Š x H) [mm]	25 x 180 x 40
Rozhranie	USB 2.0

### 1.2.3 Zivid One<sup>+</sup>

Senzor Zivid One<sup>+</sup> je, na rozdiel od senzorov Microsoft® Kinect™ 1.0 a Asus® XtrionPro™ Live, primárne určený na priemyselné použitie. Senzor sníma hĺbku priestoru využitím metódy štruktúrovaného svetla technikou projekcie binárneho kódu (bieleho svetla). Používaná technika projekcie neškodí ľudskému zraku, takže je bezpečné senzor používať v prostredí, kde sa pohybujú ľudia. [13]

Tabuľka 1.5: Základné parametre senzoru Zivid One<sup>+</sup> [14]

Rozlíšenie hĺbkového senzoru [px]	1920 x 1200
Rýchlosť snímkovania	12 S/s
Rozmery (V x Š x H) [mm]	86 x 226 x 165
Rozhranie	USB 3.0



## 1 METÓDY BEZKONTAKTNÉHO MERANIA HĽBKY

Ako bolo spomenuté v kapitole 1.2, použitím projekcie binárneho kódu ako jedného zo vzorov z kategórie sekvenčnej projekcie je možné získať hĺbkovú mapu s vysokým rozlíšením a veľkou presnosťou, avšak rýchlosť získavania snímok je pomerne nízka. Dáta získané zo senzoru sú priamo vo formáte mračna bodov (PC) obsahujúceho dáta o nameranej hĺbke, farbe a kontraste. [9][13]



Obr. 1.12: Senzor Zivid One+ [13]

Senzor je napájaný jednosmerným napätím o veľkosti 24 V so spotrebou maximálne 96 W. Odolnosť senzoru zaručuje hliníkové šasi, ktoré okrem odolnosti zaisťuje aj dobrý odvod tepla a stupeň krytia IP65. [13]

Senzor je ponúkaný v troch variantách - S, M a L. Rozdielne parametre verzií sú uvedené v nasledujúcej tabuľke.

Tabuľka 1.6: Rozdiely v parametroch senzorov Zivid One<sup>+</sup> S, M a L [14]

	S	M	L
Optimálna pracovná vzdialenosť [mm]	400	600	1200
Minimálna vzdialenosť [mm]	300	600	1200
Maximálna vzdialenosť [mm]	1000	2000	3000
Zorné pole pri min. vzdialenosti [mm]	164 x 132	433 x 271	843 x 530
Zorné pole pri max. vzdialenosti [mm]	621 x 439	1330 x 871	2069 x 1310
Najmenšia odchýlka merania [μm] *	25	60	225
Najväčšia odchýlka merania [μm] **	140	570	3680

\* Minimálna odchýlka vzdialenosti jedného bodu pri meraní na optimálnej pracovnej vzdialenosti

\*\* Maximálna odchýlka vzdialenosti jedného bodu pri meraní na maximálnej vzdialenosti a za intenzity okolitého osvetlenia  $L = 750$  lux

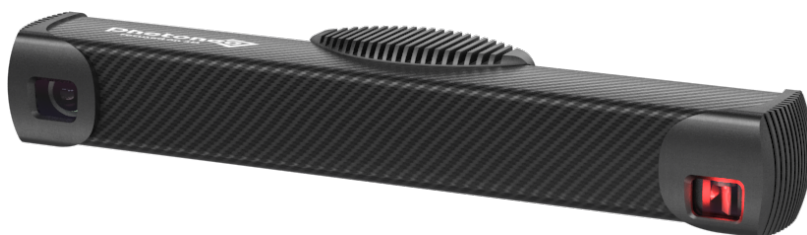
Hodnota počtu snímok za sekundu z tabuľky 1.5 je najvyššia dosiahnuteľná rýchlosť snímkovania. Reálna rýchlosť získavania hĺbkových dát v podobe mračna bodov je približne 8 snímok za sekundu a menej. [13][14]

Prácu so senzorom uľahčuje podporná knižnica pre jazyk Python. [13]



### 1.2.4 PhoXi 3D skener

Zariadenie PhoXi 3D je senzor priamo určený na skenovanie objektov. Hĺbku meria metódou štruktúrovaného svetla. Ako zdroj svetla je použitý laser. K projekcii využíva unikátnu techniku paralelného štruktúrovaného svetla vyvinutú priamo spoločnosťou Photoneo s.r.o. Technika paralelného štruktúrovaného svetla je princípom podobná sekvenčnej projekcii (postupná projekcia viacerých vzorov), no v tomto prípade sú vzory premietané súčasne. Táto technika zdieľa výhody sekvenčnej projekcie, ako je veľké rozlíšenie a vysoká presnosť, a zároveň rieši problém s časom potrebným pre zhotovenie jedného merania. Zariadením je možné snímať dynamické objekty, pričom je scéna snímaná vo vysokom rozlíšení a s veľkou presnosťou. Vďaka pokročilým metódam potlačania okolitého svetla umožňuje skener zhotoviť dostatočne kvalitné snímky aj za priameho osvetlenia scény okolitým svetlom s vysokou intenzitou.[15]



Obr. 1.13: Skener PhoXi 3D [16]

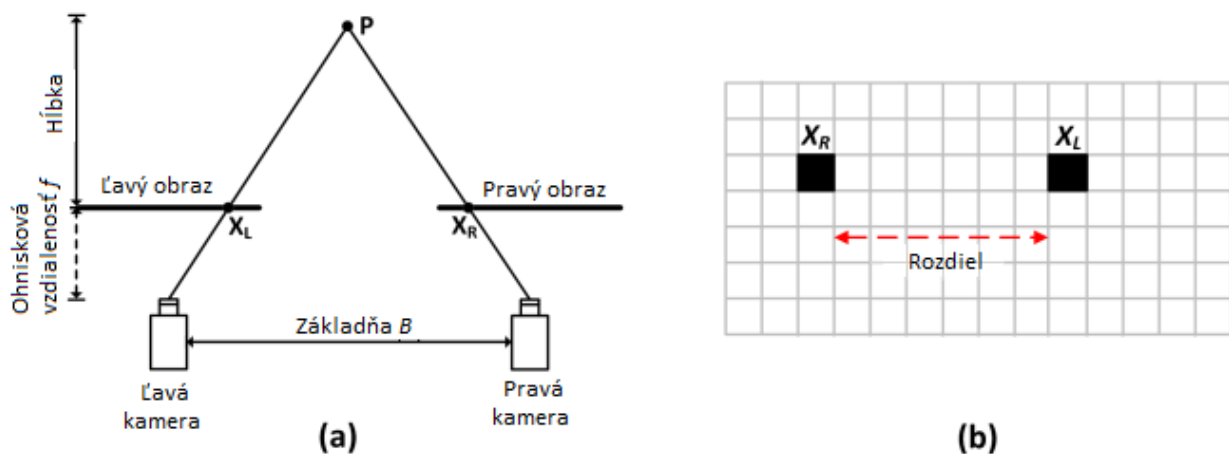
Okrem vlastného pracovného prostredia je možné so skenerom pracovať aj v prostredí LabView prostredníctvom podporného balíčku. Skenery sú dostupné v piatich variantách, a to XS, S, M, L, XL. V nasledujúcej tabuľke sú uvedené základné parametre skenerov a rozdiely medzi jednotlivými variantami. [16]

Tabuľka 1.7: Základné parametre skenerov PhoXi 3D [16]

	XS	S	M
Rozsah merateľnej vzdialenosti [mm]	161 až 205	384 až 520	458 až 1118
Snímaná plocha [mm]	113 x 83	360 x 286	590 x 421
Vzdialenosť bodov v rovine XY [mm]	0.055	0.174	0.286
Presnosť [mm]	0.035	0.05	0.1
Rozmery (V x Š x H) [mm]	77 x 296 x 68	77 x 296 x 68	77 x 416 x 68
	L	XL	
Rozsah merateľnej vzdialenosti [mm]	870 až 2150	1680 až 3780	
Snímaná plocha [mm]	1082 x 802	1954 x 1509	
Vzdialenosť bodov v rovine XY [mm]	0.524	0.947	
Presnosť [mm]	0.2	0.5	
Rozmery (V x Š x H) [mm]	77 x 616 x 68	77 x 941 x 68	

### 1.3 Aktívne a pasívne stereo snímanie

Metóda stereo snímania je odvodená z reálneho života pretvorením spôsobu vnímania priestoru ľudským zrakom na spôsob vnímania priestoru počítačom. Podobne ako ľudský mozog spracováva obrazové vnemy z oboch očí a spojením týchto dvoch vnemov vytvára priestorový obraz a je schopný vnímať a určovať hĺbkú priestoru, tak aj digitálny signálový procesor (DSP) stereoskopického senzoru je tak schopný určiť vzdialenosť jednotlivých pixelov zosnímaného obrazu, a tak vytvoriť hĺbkovú mapu. Pre vytvorenie hĺbkovej mapy sú potrebné dve kamery umiestnené najlepšie v jednej rovine so známou vzájomnou vzdialenosťou ich stredov  $B$  anglicky označovanú ako *baseline* alebo IPD - vnútro-zrenicová vzdialenosť. [1][2]



Obr. 1.14: (a) schematické zobrazenie stereo párovania, (b) poloha totožného bodu v snímke pravej a ľavej kamery [17]

Keďže sú kamery navzájom od seba vzdialené, každá z nich zachytáva scénu z iného uhlu. K zisteniu hĺbkovej je potrebné získať rozdiely polôh totožných bodov, ako je vidno na obrázku 1.14 (a), kde  $x_L$  a  $x_R$  sú priemet bodu  $P$ . Po spojení oboch snímok do jednej a vyobrazení priemetov  $x_L$  a  $x_R$  bodu  $P$  je vidno rozdiel v ich polohe, vďaka čomu je možné použitím vzťahu 1.5 určiť vzdialenosť bodu  $P$  od senzoru. Vo vzťahu 1.5 je  $f$  ohnisková vzdialenosť kamier,  $B$  je vzájomná vzdialenosť stredov kamier,  $x_R$  a  $x_L$  sú polohy priemetov bodu  $P$ . Maximálnu a minimálnu vzdialenosť merania je možné upraviť zmenou vzájomnej vzdialenosti kamier - čím menšia je vzdialenosť kamier, tým menšia je minimálna merateľná vzdialenosť. Treba však dbať na maximálnu merateľnú vzdialenosť. V okamihu, kedy je meraný bod od senzoru vzdialený natoľko, že je priemet tohto bodu v ľavom aj pravom obraze na rovnakej pozícii, čiže  $x_R - x_L = 0$ , je, ako vyplýva zo vzťahu 1.5, vzdialenosť bodu vyhodnotená ako nekonečno. [2][3][17]

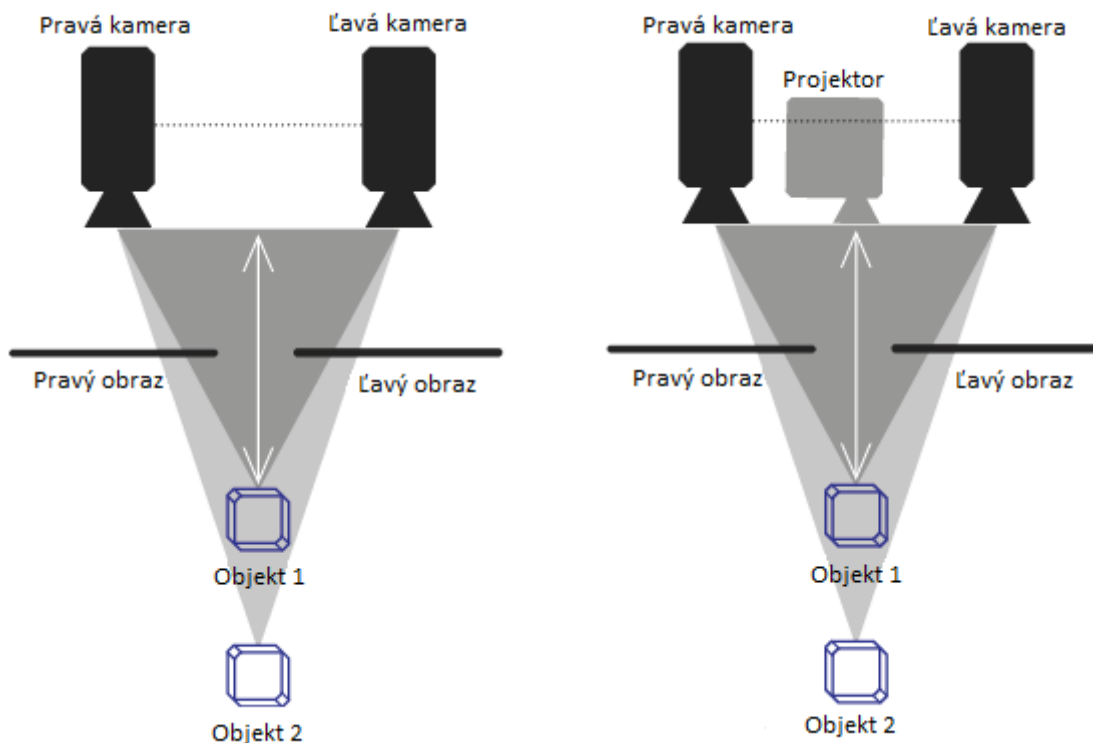
$$D = \frac{B \cdot f}{x_R - x_L} \quad (1.5)$$

Vyššie je popísaná metóda pasívneho stereo snímania. Táto metóda je hardwarovo nenáročná, sú potrebné iba dve, najlepšie totožné, kamery. Pasívne stereo snímanie je však plne závislé na okolitom osvetlení, keďže pasívne stereo senzory nemajú zakomponovaný žiaden zdroj svetla. Taktiež z hľadiska vyhľadávania zhodných pixelov v dvoch obrazoch nastáva problém pri snímaní hladkých jednofarebných povrchov.[1]

Problém so snímaním jednofarebných hladkých povrchov rieši metóda aktívneho stereo snímania, ktorá kombinuje pasívne stereo snímanie a metódu štruktúrovaného svetla. Aktívne stereo

## 1 METÓDY BEZKONTAKTNÉHO MERANIA HĽBKY

snímače majú okrem dvoch kamier vstavaný projektor premietajúci statický vzor. Premietanie vzoru na snímanú scénu umožňuje stereo snímaču jednoduchšie určiť priemety bodu v snímkach.[2]



Obr. 1.15: Metóda pasívneho stereo snímania (vľavo) a aktívneho stereo snímania (vpravo) [2]

Výhodou metód je jednoducho pochopiteľný princíp, keďže je podobná ľudskému zraku. Obe metódy majú dobré rozlíšenie v rovine XY, toto rozlíšenie závisí najmä od rozlíšenia kamier. Taktiež cena zariadení pre implementáciu metódy stereo snímania je nízka. V prípade aktívneho stereo snímania však ide komplexné riešenie, keďže narozdiel od ostatných metód implementuje okrem vysielачa a snímača ešte jeden snímač. Metóda stereo snímania je náročná na softwarové riešenie. [1]

### 1.3.1 MultiSense S7

Hĺbkový senzor od spoločnosti Carnegie Robotics merajúci vzdialenosť využitím metódy pasívneho stereo snímania. Pri kúpe je možná voľba medzi farebným a čiernobielym senzorom a objektívom s užším alebo širším zorným polom. Dáta zo senzoru sú prenášané pomocou ethernetovej prípojky. Napájaný je jednosmerným napätím o veľkosti 24 V a jeho maximálna spotreba dosahuje hodnotu 20 W. So stupňom krytia IP68 je senzor vhodný pre použitie v prašnom a vlhkom prostredí, taktiež je odolný voči ponoreniu do vody. Práca so senzorom je možná použitím voľne dostupnej knižnice pre jazyk C++. [18]



Obr. 1.16: Senzor MultiSense S7 [18]

Tabuľka 1.8: Základné parametre senzoru MultiSense S7 [7]

Rozlíšenie hĺbkového senzoru [px]	2048 x 1088
Rozlíšenie farebnej kamery [px]	2048 x 1088
Merateľná vzdialenosť [m]	0.4 až nekonečno
Rýchlosť snímkovania [S/s]*	15
Zorné pole (H x V)	80° x 45°
Rozmery (V x Š x H) [mm]	65 x 130 x 130
Rozhranie	Gigabitový ethernet

\* Rýchlosť snímkovania pri rozlíšení 2048 x 544

### 1.3.2 ZED

Stereoskopický senzor vyvinutý spoločnosťou Stereolabs merajúci hĺbku metódou pasívneho stereo snímania. Je možné vybrať si z troch modelov senzoru, a to ZED, ZED 2 a ZED Mini, parametre a rozdiely modelov sú uvedené v tabuľke 1.9.



Obr. 1.17: Senzory ZED (hore), ZED 2 (v strede) a ZED Mini (dole) [19][20][21]

Vysoká rýchlosť snímania (pri menšom rozlíšení až 100 S/s) umožňuje senzorom snímať dynamicky sa meniace okolie. Modely ZED 2 a ZED Mini sú, vďaka integrovaným senzorom pohybu a polohy (akcelerometer a gyroskop), vhodné pre aplikácie využívajúce metódu simultánnej lokalizácie a mapovania (SLAM). Metóda SLAM je často využívaná pri návrhu autonómnych robotov, vozidiel a dronov, ktoré sú samostatne schopné orientovať sa vo svojom okolí a zároveň ho mapovať, alebo pri zmiešanej virtuálnej realite. Senzor ZED 2 má navyše vstavaný barometer, magnetometer a teplotné čidlo umožňujúce kompenzáciu vplyvov teploty, tlaku a magnetického poľa na meranie. [20]

Tabuľka 1.9: Parametre senzorov ZED, ZED 2 a ZED Mini [19][20][21]

	ZED	ZED 2	ZED Mini
Maximálne rozlíšenie obrazu [px]	4416 x 1242	4416 x 1242	4416 x 1242
Rýchlosť snímokovania [S/s] *	100	100	100
Rozsah hĺbky [mm]	0.3 až 25	0.2 až 20	0.1 až 15
Zorné pole (H x V x D)	90° x 60° x 100°	110° x 70° x 120°	90° x 60° x 100°
Rozmery (V x Š x H) [m]	30 x 175 x 33	30 x 175 x 33	26.5 x 30.5 x 124.5
Pohybové a polohové senzory	N/A	Akcelerometer	Akcelerometer
		Gyroskop	Gyroskop
		Barometer	
		Magnetometer	
		Teplotný senzor	
Rozhranie	USB 3.0	USB 3.0	USB 3.0 typ C

\* Rýchlosť snímania pri rozlíšení 1344 x 376 pixelov

Senzory sú dodávané s balíčkami knižníc pre jazyky ako je Matlab, Python, C++, OpenCV a pod. [20]

### 1.3.3 Intel® RealSense™

Spoločnosť Intel® rozšírila radu svojich produktov o snímače hĺbký osadené čipmi vlastnej výroby. Ich aktuálne vlajkové lode merajúce vzdialenosť metódou stereo snímania sú kamery rady D400. Okrem senzorov pracujúcich na princípe stereo snímania sú v ponuke aj senzory pracujúce na princípe štruktúrovaného svetla, rada SR300, senzor využívajúci k meraniu metódu LIDaR, rada L500, a senzor T265 merajúci vzdialenosť farebným stereoskopickým modulom využívajúc metódu SLAM. [22][23][24]



Obr. 1.18: Senzory RealSense™ D415 (hore), D435 a D435i (dole) [22][23]

Jadrom senzorov rady D400 je stereo modul snímajúci hĺbku metódou aktívneho stereo snímania. Kvalita priestorového obrazu je zlepšená využitím projekcie infračerveného vzoru s nastaviteľnou intenzitou na snímanú scénu. Vďaka projekcii infračerveného vzoru a snímaniu v IR spektre je možné senzory použiť aj v horších svetelných podmienkach. Senzor D435i, narozdiel od modelu D435 a D415, obsahuje vnútornú meraciu jednotku (IMU) so zabudovaným akcelerometrom a gyroskopom. [22][23][24] Pre umožnenie práce so senzorom sú dostupné podporné

Tabuľka 1.10: Základné parametre senzorov Intel® RealSense™ D415, D435 a D435i [22][23][24]

	D415	D435	D435i
Rozlíšenie hĺbkového senzoru [px]	1280 x 720		
Rozlíšenie farebného senzoru [px]	1920 x 1080		
Rýchlosť snímokovania [S/s] *	90/30		
Rozsah vzdialenosti [m]	0.16 až 10	0.105 až 10	0.105 až 10
Zorné pole (H x V x D) **	65° x 40° x 72°	87° x 58° x 95°	87° x 58° x 95°
	69.4° x 42.5° x 77°		
Rozmery (V x Š x H) [mm]	20 x 99 x 23	25 x 90 x 25	25 x 90 x 25
Druh uzávierky	Roletová	Globálna	Globálna
Rozhranie	USB-C Gen 3.1		

\* Rýchlosť snímania hĺbkového/farebného senzoru

\*\* Zorné pole hĺbkového/farebného senzoru

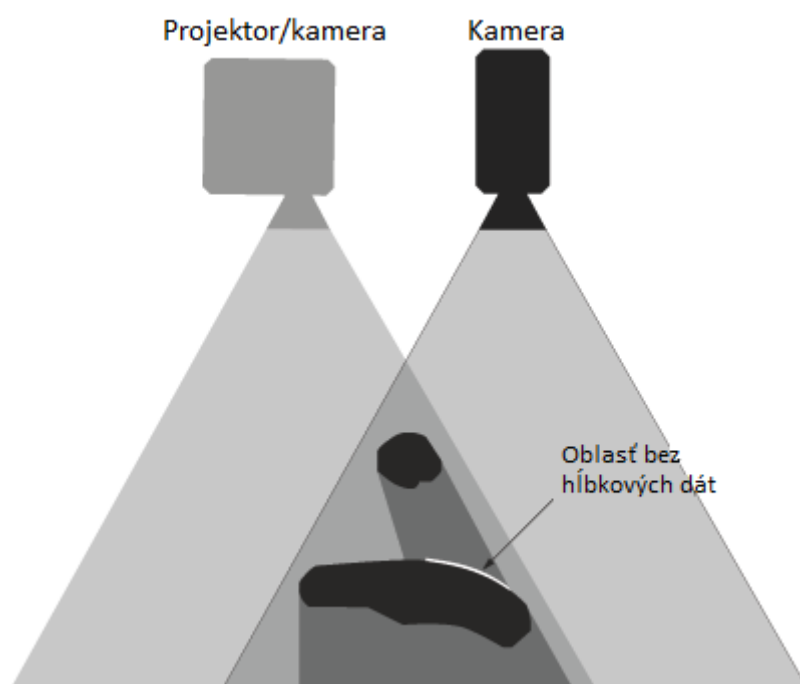
balíčky pre mnohé programovacie jazyky ako je C++, Python, Matlab a pod. [22]

Senzor RealSense D415 je použitý v praktickej časti tejto práce, kde je princíp jeho funkcie bližšie popísaný.

## 1.4 Zhrnutie a porovnanie metód merania hĺbky

Každá z vyššie popísaných metód má svoje výhody a nevýhody. Žiadna z nich nie je univerzálna, kľúčom k úspechu je voľba vhodnej metódy podľa konkrétnych potrieb a podmienok.

Jediná z vyššie spomenutých metód merajúca vzdialenosť priamo je metóda TOF. V porovnaní s ostatnými metódami, pri meraní metódou TOF nevzniká efekt zatienenia. [1]



Obr. 1.19: Znázornenie efektu zatienenia [2]

Efekt zatienenia, ako je možné vidieť na obrázku 1.19, v prípade metódy SL vzniká premietaním vzoru z iného uhlu než je kamerou obraz zachytávaný, čiže objekty v popredí vrhajú tieň na objekty za nimi z pohľadu paprskov vychádzajúcich z projektoru aj napriek tomu, že z pohľadu kamery je dobre vidieť aj zatienená časť scény. Pri metóde stereo snímania (na obr. 1.19 je projektor nahradený kamerou) je síce obraz snímaný obomi kamerami, ale vo výsledku je snímaný obraz premietnutý do pohľadu jednej z kamier, pričom dáta z druhej kamery slúžia pre určenie rozdielu v polohe bodov. Efekt zatienenia je dobre pozorovateľný najmä pri meraní na veľmi krátku vzdialenosť, kedy je rozdiel medzi časťami scény, ktoré vidia jednotlivé kamery, markantný. Z toho vyplýva, že sa vplyv efektu zatienenia na snímaný hĺbkový obraz s rastúcou vzdialenosťou znižuje. [1][2]

Ďalšou výhodou metódy TOF voči ostatným metódam je jej takmer úplná nezávislosť na okolitom osvetlení, nezávislosť na štruktúre, drsnosti a farbe snímaného objektu a vysoká rýchlosť merania. [1]

Nevýhodou metódy TOF v porovnaní s ostatnými metódami je hlavne relatívne nízke rozlíšenie senzoru a tým malá presnosť hodnôt v rovine XY a efekt viaccestného odrazu, kedy odrazený lúč nemusí smerovať priamo k senzoru, ale môže sa počas návratu viackrát odraziť. [1][2]

Vysoké rozlíšenie a veľká presnosť merania hĺbkových rozmerov, ale aj rozmerov v rovine XY, sú hlavnou prednosťou metódy SL. Použitím tejto metódy je možné získať najpresnejšie hodnoty

## 1 METÓDY BEZKONTAKTNÉHO MERANIA HLĚBKY

merania, avšak za cenu rýchlosti získavania dát. Táto metóda je vhodná najmä pre snímánie pomaly sa meniacej alebo statickej scény. Taktiež okolité osvetlenie a drsnosť povrchu zohrávajú dôležitú rolu pri meraní hĺbky touto metódou. Princípom metódy je projekcia predom definovaného vzoru, čiže si senzory samé osvetľujú snímanú scénu. Metóda je vhodnejšia pre použitie v slabšie osvetlených priestoroch, ako sú napríklad interiéry budov. [1][2]

Metódy merania hĺbky stereo snímaním sú jednoduché na pochopenie, keďže sú podobné princípu priestorového videnia človeka. Metóda pasívneho stereo snímania je hardwarovo nenáročná, avšak vyžaduje zložitejšie vyhľadávacie a párujúce algoritmy. Stereo snímanie je vhodné pre meranie hĺbky pri veľkej intenzite okolitého svetla, avšak je nevhodné pre snímánie jednofarebných objektov s nevýraznou štruktúrou. Tento problém rieši metóda aktívneho stereo snímania, kedy je na snímaný objekt premietaný vzor, ktorý napomáha priradeniu správnych pixelov k sebe. Ďalšou výhodou aktívneho stereo snímania je využitie snímačov pracujúcich v IR spektre, vďaka čomu zaniká úplná závislosť na okolitom svetle a je možné merať hĺbku aj v šere. Podstatnou nevýhodou metódy stereo snímania oproti ostatným metódam je kvadratická závislosť chyby merania na vzdialenosti. [1][2][3]

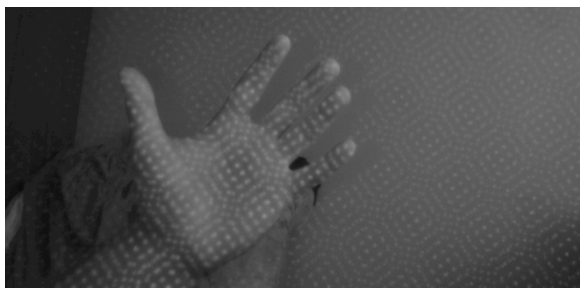
Tabuľka 1.11: Porovnanie metód snímania hĺbkovej mapy [1][2][3][7]

	TOF	SL		Stereo snímanie	
		Meniaci sa vzor	Stály vzor	Aktívne	Pasívne
Hĺbková presnosť	mm-cm	μm-mm	mm-cm	mm-cm	mm-cm
Presnosť v rovine XY	Malá	Veľká	Stredná	Stredná	Stredná
Rýchlosť snímkovania	Rýchla	Pomalá	Rýchla	Stredná	Stredná
Merateľná vzdialenosť	Malá - Veľká	Malá - Stredná	Malá - Stredná	Stredná	Stredná
Kvalita v slabom svetle	Dobrá	Dobrá	Dobrá	Dobrá	Zlá
Kvalita v silnom svetle	Dobrá	Zlá	Zlá	Dobrá	Dobrá
Spotreba energie	Stredná	Stredná	Stredná	Nízka	Stredná
Hardwarové náklady	Stredné	Vysoké	Vysoké	Vysoké	Nízke
Softwarová náročnosť	Nízka	Vysoká	Stredná	Vysoká	Vysoká
Farba snímaného objektu	Nezáleží	Nezáleží	Nezáleží	Nezáleží	Záleží
Štruktúra povrchu	Nezáleží	Záleží	Záleží	Nezáleží	Záleží



## 2 Intel® RealSense™ rady D400

Kamery RealSense rady D400 využívajú k meraniu vzdialenosti metódu aktívneho stereo snímania snímajúc scénu dvomi infračervenými modulmi. Meraniu hĺbky napomáha laserový projektor premietajúci opakujúci sa vzor v infračervenom spektre na sledovanú scénu. Projekcia vzoru zvyšuje presnosť merania vzdialenosti a napomáha meraniu pri snímaní jednofarebných objektov s nízkou štruktúrou, ako je napríklad biela stena, ako je možné vidieť na obrázku 2.1. Spolu s dvojicou infračervených snímacích modulov je v senzore implementovaná kamera snímajúca viditeľné spektrum.



Obr. 2.1: Infračervená snímka z modelu D415 s viditeľným premietaným vzorom

Rozdielom medzi modelom D415 a modelmi D435 a D435i je ich zorné pole. Model D415 je osadený kamerovými modulmi so štandardnými objektívmi a modely D435 a D435i sú osadené kamerovými modulmi so širokouhlými objektívmi. Model D435i má navyše k dispozícii gyroskop a akcelerometer zaznamenávajúce polohu a pohyb zariadenia, čo umožňuje využiť tento model pre aplikácie ako je mapovanie okolia, napr. metódou SLAM. [25]

Ako bolo spomenuté v kapitole 1.3, pri metóde stereo snímania je vzdialenosť bodu (pixelu) určená ako rozdiel v polohe daného bodu na snímkach z kamier snímajúcich scénu. Maximálna a minimálna merateľná hĺbka sa mení s meniacou sa vzájomnou vzdialenosťou kamier (IPD). IPD modelu D415 je približne 55 mm, čo určuje minimálnu merateľnú vzdialenosť na približne 430 mm od kamery. Teoretická maximálna merateľná vzdialenosť je nekonečno, avšak rozdiel polôh bodu v určitej vzdialenosti je na snímkach z kamier nulový, čím je vytvorená hranica maximálnej merateľnej vzdialenosti. Z toho vyplýva, že maximálna merateľná vzdialenosť je taktiež závislá na nastavenom rozlíšení. Takto určená maximálna merateľná vzdialenosť však nie je vždy plne užitočná, keďže chyba merania je závislá na vzdialenosti a rastie s druhou mocninou. V istom bode je chyba merania natoľko veľká, že sa meranie nedá naďalej považovať za dostatočne presné. Z toho dôvodu je maximálna merateľná vzdialenosť kamier RealSense rady D400 výrobcom stanovená na 10 m pri rozlíšení 1280x720 pixelov pre model D415 a 848x480 pixelov pre modely D435 a D435i. [25]

Hĺbka je vypočítavaná procesorom D4 - mikroprocesor vyvinutý špeciálne pre účely spracovania obrazu (VPU), ktorý je implementovaný priamo v kamere. Výstupom je 16-bitová hodnota. Presnosť merania je meniteľná, spôsob jej menenia je popísaný v nasledujúcej kapitole. [26]

## 3 Komunikačný protokol

K zariadeniam Intel RealSense sú výrobcom vytvorené podporné balíky (známe aj ako "wrappre") umožňujúce prácu s kamerou v rôznych jazykoch. Podporné balíky sa dajú získať pri inštalácii programu *Intel RealSense SDK 2.0*, kde je v jednom kroku inštalácie možnosť stiahnuť balíky pre jazyky ako je MATLAB, Python, C++ atď. Druhá z možností ako ich získať je stiahnuť ich priamo z platformy *github*.

### 3.1 Podporný balík pre prostredie MATLAB

Po získaní balíku je potrebné pridať cestu k nemu do prostredia MATLAB. Návod pre trvalé pridanie cesty k podpornému balíku je v ukážke 3.1.

```
% podporny balicek +realsense je ulozeny v priecinku Matlab na disku C

addpath('C:\Matlab\');      % prida cestu k balicku
savepath;                  % ulozi cestu, pri dalsom spusteni programu
                             % nie je potrebne cestu znovu pridavat
```

Ukážka 3.1: Trvalé pridanie cesty k podpornému balíčku

Základom podporného balíku je skompilovaný kód pre ovládanie zariadenia Intel RealSense vo formáte *.mexw64*, čo znamená, že je balík možné použiť iba v 64 bitovej verzii Windowsu.

Okrem skompilovaného kódu je v zložke množstvo *.m* skriptov, konkrétne objektov slúžiacich na ovládanie a komunikáciu so zariadením. Každý z objektov má vlastné metódy, pomocou ktorých je možné meniť nastavenia zariadenia, získavať dáta a pod. Objekty sú navzájom prepojené a spolu vytvárajú jeden celok - ovládač. Používanie tohto balíka je značne obtiažne. Objekty v balíčku je možné rozdeliť na triedy a podtriedy, čo môže spôsobiť značný chaos pri snahe komunikovať so zariadením. Ďalším problémom je absencia poznámok a popisov jednotlivých metód. Taktiež potreba inicializácie a prvotných nastavení zariadenia nie je dostatočne zrozumiteľná a často môže ústiť k zbytočným chybám v kóde. Tieto fakty sú dôvodom pre vytvorenie jednoduchšieho komunikačného protokolu vo forme objektu, ktorý umožňuje užívateľovi jednoduchšiu komunikáciu so zariadením RealSense.

## 3.2 Objekt RealSense

Po získaní podporného balíčka bolo potrebné prejsť jednotlivé m-skripty a skontrolovať chyby v syntaxi ako napr. chýbajúce alebo prevyšujúce *end* a taktiež chyby v kontrole argumentov zadávaných do jednotlivých metód. Tieto chyby v podpornom balíčku spôsobovali počiatočné potiaže pri vytváraní nového komunikačného protokolu a museli byť odstránené. Jednou z príloh tejto práce je archív obsahujúci opravený podporný balíček. Pri práci s komunikačným protokolom popísaným v tejto práci je odporúčané používať práve priložený podporný balíček. Pri použití iného podporného balíčka nie je zaručené správne fungovanie komunikačného protokolu.

Ako alternatívny spôsob komunikácie so zariadeniami Intel RealSense rady D400 je objekt RealSense vytvorený v prostredí MATLAB. Objekt RealSense (RSO) spája jednotlivé metódy podporného balíčka do celkov a poskytuje užívateľovi jednoduchší a intuitívnejší spôsob práce s kamerou. Pri práci s kamerou je vytvorené veľké množstvo rôznych objektov, bez ktorých by nebolo možné s kamerou komunikovať. Tieto objekty sú spolu s ďalšími dôležitými premennými ukladané do parametrov RSO. Pre zaistenie bezchybnej funkčnosti pri práci s RSO sú parametre objektu rozdelené do troch skupín.

Parametrami prvej skupiny sú objekty vytvorené konštruktérom alebo metódou *start()* a prvky grafického užívateľského prostredia (GUI) panelu nastavení. Keďže správna funkčnosť RSO závisí na týchto parametroch, je ich dostupnosť nastavená na *private*, čo znamená, že po vytvorení RSO má k týmto parametrom prístup iba samotný objekt a ďalšie podobъекty, ale užívateľ k nim priamy prístup nemá. Potrebné parametre a metódy objektov v tejto skupine sú dostupné skrz metódy RSO.

Druhú skupinu tvoria parametre prístupné užívateľovi iba na čítanie. Prístup k týmto parametrom je nastavený na *immutable*, čo značí, že sú nemenné. Majú informatívny charakter. Obsahujú informácie o dostupných filtroch, nastaveniach, informácii o zariadení a pod. RSO s nimi pracuje pri vyhodnocovaní validity argumentov, ktoré užívateľ zadáva pri vyvolaní niektorej metódy RSO.

Poslednú skupinu tvoria verejné parametre, prístup k nim je nastavený na *public*, čo značí, že po vytvorení RSO k nim majú prístup iné objekty, funkcie a užívateľ. Tieto parametre sú vytvorené za rôznymi účelmi. Niektoré majú informatívny charakter, ako napr. štruktúra *Stream\_info* obsahujúca informácie o aktívnych dátových tokoch alebo parameter *Frames\_count* obsahujúci informáciu o počte získaných snímok od začiatku dátového toku. Ďalšie parametre v tejto skupine majú charakter nastavení, ako napr. parameter *Align* meniaci nastavenie filtra, ktorý zodpovedá za zarovnanie farebnej a hĺbkovej snímky alebo parametrov *Depth\_resolution*, *Color\_resolution* a *Infrared\_resolution* zodpovedajúcich za nastavenie rozlíšenia a frekvencie snímkovania jednotlivých typov snímok.

Takto roztriedené parametre umožňujú pracovať s RSO bez rizika nechcenej zmeny dôležitých parametrov, čo by mohlo spôsobiť závažnú chybu programu, a zároveň poskytujú užívateľovi potrebné informácie pre prácu s RSO.

Ďalšou časťou RSO sú metódy. Metódy objektu sú funkcie deklarované a definované v objekte. Majú oprávnenie pracovať so všetkými tromi skupinami parametrov popísanými vyššie. Metódy sú taktiež rozdelené do skupín, a to na verejné a súkromné metódy. Verejné metódy podobne ako verejné parametre môže užívateľ používať. Slúžia k inicializácii zariadenia, zmene nastavení a k získavaniu dát z kamery. Na druhú stranu súkromné metódy užívateľ nevidí a nemá k nim prístup. Tieto metódy sú pomocné a tak sú využívané verejnými metódami v prípadoch, kedy

sa dané príkazy opakujú vo viacerých metódach a je možné vytvoriť univerzálne funkciu, ktorú môžu využiť ostatné metódy podľa potreby. Ďalšími metódami v tejto skupine sú metódy vytvárajúce GUI a pracujúce v pozadí pri používaní daného GUI.

Všetky užívateľovi dostupné metódy majú stručný popis obsahujúci popis funkcie, vstupy a výstupy s ich popisom a formátom dát.

### 3.2.1 Inicializácia objektu RealSense

Pre vytvorenie akéhokoľvek objektu v prostredí MATLAB je používaná metóda všeobecne známa ako konštruktér. Konštruktér je metóda alebo funkcia, ktorej názov je zhodný s názvom objektu. Metóda môže mať ľubovoľný počet vstupov a jeden výstup. Výstupom konštruktéra je takzvaný *object handle*, teda odkaz (referencia) na miesto v pamäti kde je daný objekt vytvorený. Po vytvorení odkazu na RSO získa užívateľ prístup k verejným parametrom a metódam.

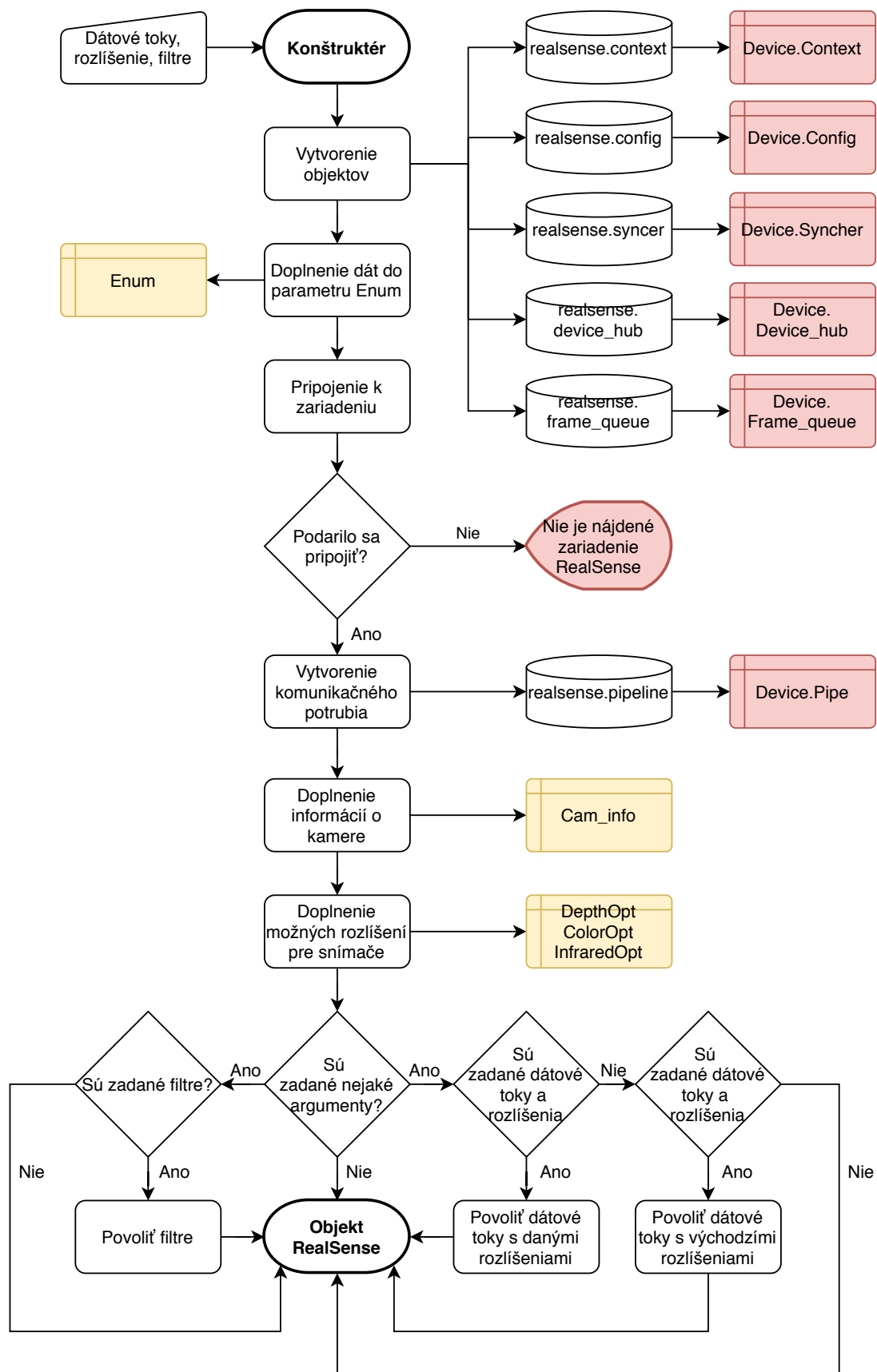
Konštruktér okrem RSO vytvorí ďalšie objekty a doplní dáta do premenných ako je vidno v diagrame na obrázku 3.1.

Pri inicializácii môže užívateľ definovať priamo dátové toky, ich rozlíšenie a filtre, ktoré chce aktivovať. Pokiaľ je zadaný ktorýkoľvek z týchto argumentov, je potrebné zadať všetky. V prípade, že počet zadaných argumentov je väčší ako nula a menší ako tri nastane chyba a RSO sa neinicializuje. Dátové toky sú zadávané vo formáte radu reťazcov (znakov spojených do jedného slova) alebo radu buniek obsahujúcich názvy dátových tokov, napr. `["color", "depth"]` alebo `{'color', 'depth'}`. Pokiaľ je zadané rozlíšenie, je potrebné zadať rozlíšenia pre všetky povolené dátové toky z predchádzajúceho argumentu. Argument rozlíšenia bude mať teda tvar matice o veľkosti  $M \times 3$ , kde  $M$  je počet zadaných dátových tokov. Stĺpce matice reprezentujú parametre šírka, výška a frekvencia snímkovania. Pokiaľ je argument rozlíšenia prázdny, teda je zadaný ako `[]`, sú pri povolení dátových tokov použité východzie hodnoty rozlíšenia a frekvencie snímkovania. Východzie hodnoty týchto parametrov sú pre všetky dátové toky nastavené na výrobcom odporúčané rozlíšenie 1280x720 pixelov a snímkovaciu frekvenciu 30 S/s. Pokiaľ je argument dátových tokov prázdny, ale je zadané rozlíšenie, nič sa nestane, rozlíšenia jednotlivých dátových tokov zostanú nastavené na východziu hodnotu, keďže nie je definované, pre ktorý dátový tok je dané rozlíšenie určené. Pokiaľ nie je argument označujúci filtre, ktoré sa pri inicializácii majú povoliť, prázdny, sú vytvorené objekty pre jednotlivé filtre zapísané v argumente. Dáta pre argument sú zadávané v rovnakom tvare ako pre argument dátové toky, teda rad reťazcov alebo buniek.

Pokiaľ užívateľ pri inicializácii nezadá žiaden argument, je RSO vytvorený bez povolených dátových tokov a filtrov, rozlíšenie pre jednotlivé dátové toky je ponechané na východzích hodnotách.

Rozlíšenie dátových tokov a povolenie filtrov a dátových tokov je možné urobiť v ďalšom kroku pri nastavení zariadenia. Legenda pre prácu s diagrammi sa nachádza v prílohách A1.

### 3 KOMUNIKAČNÝ PROTOKOL



Obr. 3.1: Diagram popisujúci činnosť konštruktéru

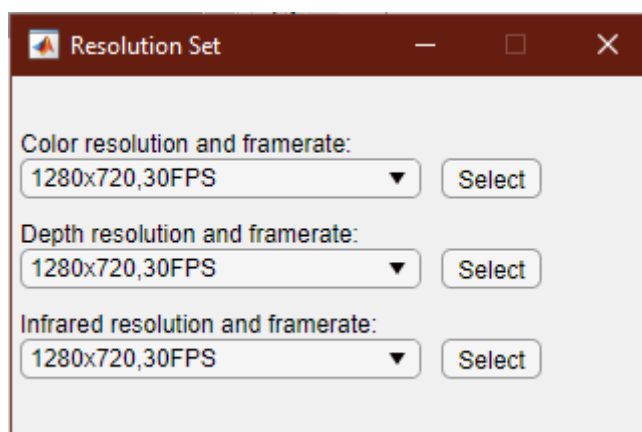
### 3.2.2 Nastavenie a spustenie dátového toku

Po vytvorení RSO má užívateľ plný prístup k metódam a parametrom umožňujúcim komunikáciu s kamerou. Pri vytváraní RSO je vytvorený objekt typu *realsense.config* uložený v chránenom parametri *Device.Config*. Tento objekt obsahuje dôležité informácie o povolených dátových tokoch potrebné pre ich správne spustenie. Vo východnom nastavení objektu je povolený hĺbkový a farebný dátový tok, oba s rozlíšením 1280x720 pixelov a frekvencií snímania 30 S/s.

Parametri jednotlivých dátových tokov (rozlíšenie a frekvencia snímokovania) je možné zmeniť tromi spôsobmi. Prvý zo spôsobov je priama zmena verejných parametrov *Color\_resolution*, *Depth\_resolution* a *Infrared\_resolution*. Tieto parametre sú zložené z radu hodnôt v tvare *[šírka, výška, frekvencia snímokovania]*. Tento spôsob je nebezpečný a môže viesť k zlyhaniu a ukončeniu programu chybovou hláškou, pokiaľ dátový tok nepodporuje dané rozlíšenie.

Druhý zo spôsobov je bezpečnejší a vhodný pre programové použitie. Rozlíšenie je možné zmeniť použitím metódy *set\_resolution()*. Argumentmi funkcie sú senzor, ktorého rozlíšenie sa má zmeniť, a hodnota, na ktorú sa má zmeniť. Hodnota môže byť zadaná dvomi spôsobmi, zadáním ako rad hodnôt v tvare *[šírka, výška, frekvencia snímokovania]* alebo ako reťazca znakov v tvare *"{šírka}x{výška},{frekvencia snímokovania}FPS"*. Zadané rozlíšenie je porovnané s podporovanými rozlíšeniami daného dátového toku. Pokiaľ sa nezhoduje so žiadnym z nich, je ponechané predchádzajúce nastavenie.

Tretou možnosťou je použitie GUI určeného pre výber rozlíšení jednotlivých dátových tokov, ktoré je možné vidieť na obrázku 3.2. Po vyvolaní metódy *open\_resolution\_tab()* je skript pozastavený, uveznávaný vo *while* slučke. Výberom požadovaného rozlíšenia a frekvencie snímokovania z ponuky pre daný dátový tok a kliknutím na tlačidlo *Select* vedľa danej ponuky je rozlíšenie a frekvencia snímokovania daného toku definitívne zmenená na vybranú hodnotu. Slučka *while* je ukončená zavretím okna GUI, skript následne pokračuje ďalej.



Obr. 3.2: GUI slúžiace na nastavenie rozlíšenia

Pre povolenie alebo zakázanie dátových tokov sú určené metódy *enable\_stream()* a *disable\_stream()*. Argumentom týchto metód je rad buniek alebo reťazcov obsahujúci dátové toky, ktoré majú byť povolené alebo zakázané. Pokiaľ argument nie je zadáný, sú povolené alebo zakázané všetky dátové toky.

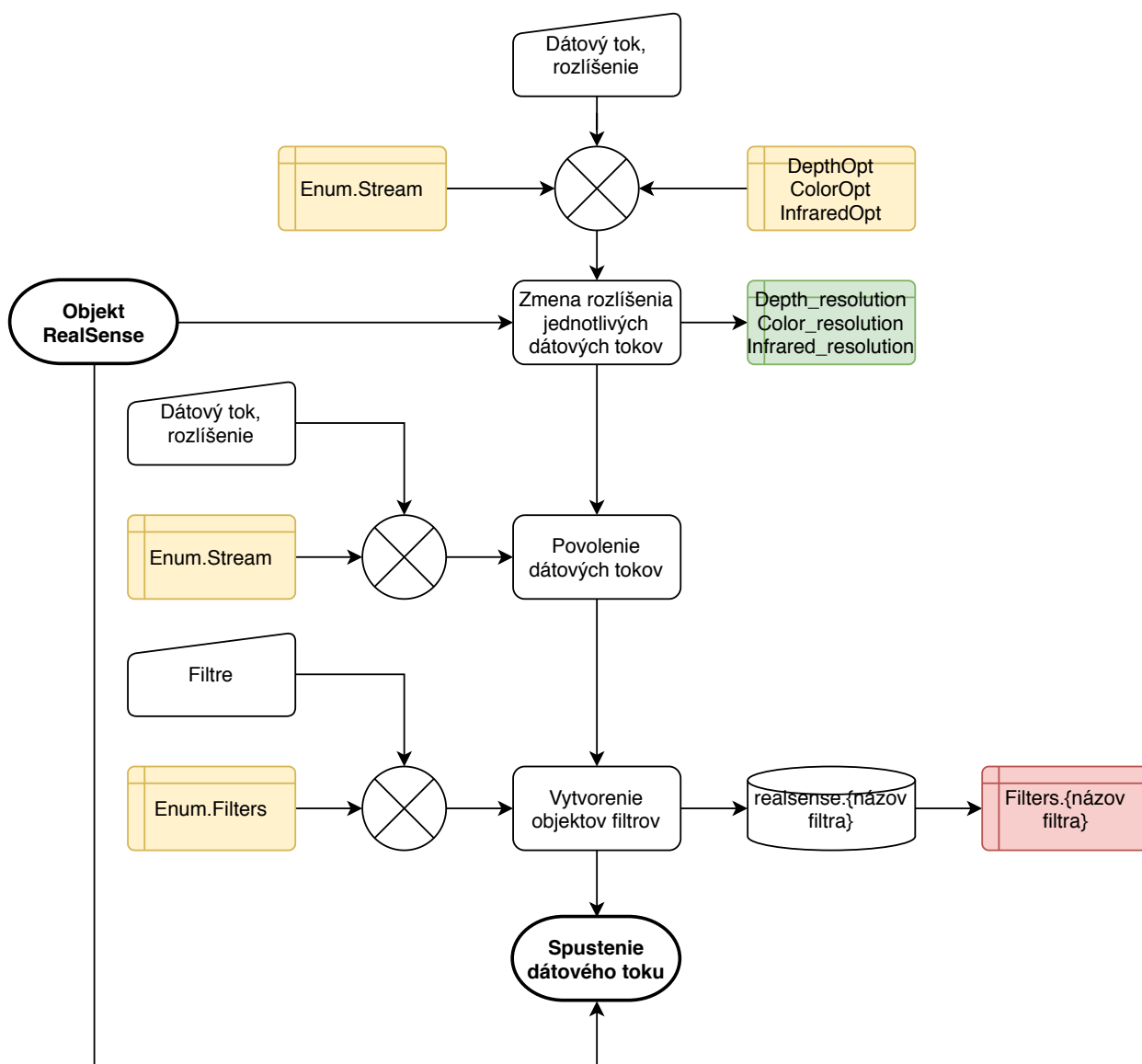
Model D415 podporuje tri druhy dátových tokov. Farebný dátový tok je získavaný z farebného modulu a hĺbkový tok z hĺbkového modulu prekrytím snímok z infračervených kamier.

### 3 KOMUNIKAČNÝ PROTOKOL

Keďže má zariadenie dve infračervené kamery, je možné získať dva infračervené toky. Ako bolo spomenuté v kapitole 1.4, pri meraní hĺbky metódou stereo snímania vzniká efekt zatienenia spôsobený vytvorením a prekrytím snímok z dvoch kamier, ktoré sú od seba navzájom vzdialené. V prípade modelu D415 je hĺbkový obraz získavaný z pohľadu ľavej infračervenej kamery, čo má za následok vznik efektu zatienenia vždy z ľavej strany od objektu v popredí.

Dátové toky je potrebné inicializovať a nastaviť pred spustením dátového toku. Po jeho spustení sa zmeny v nastaveniach už neprejavujú.

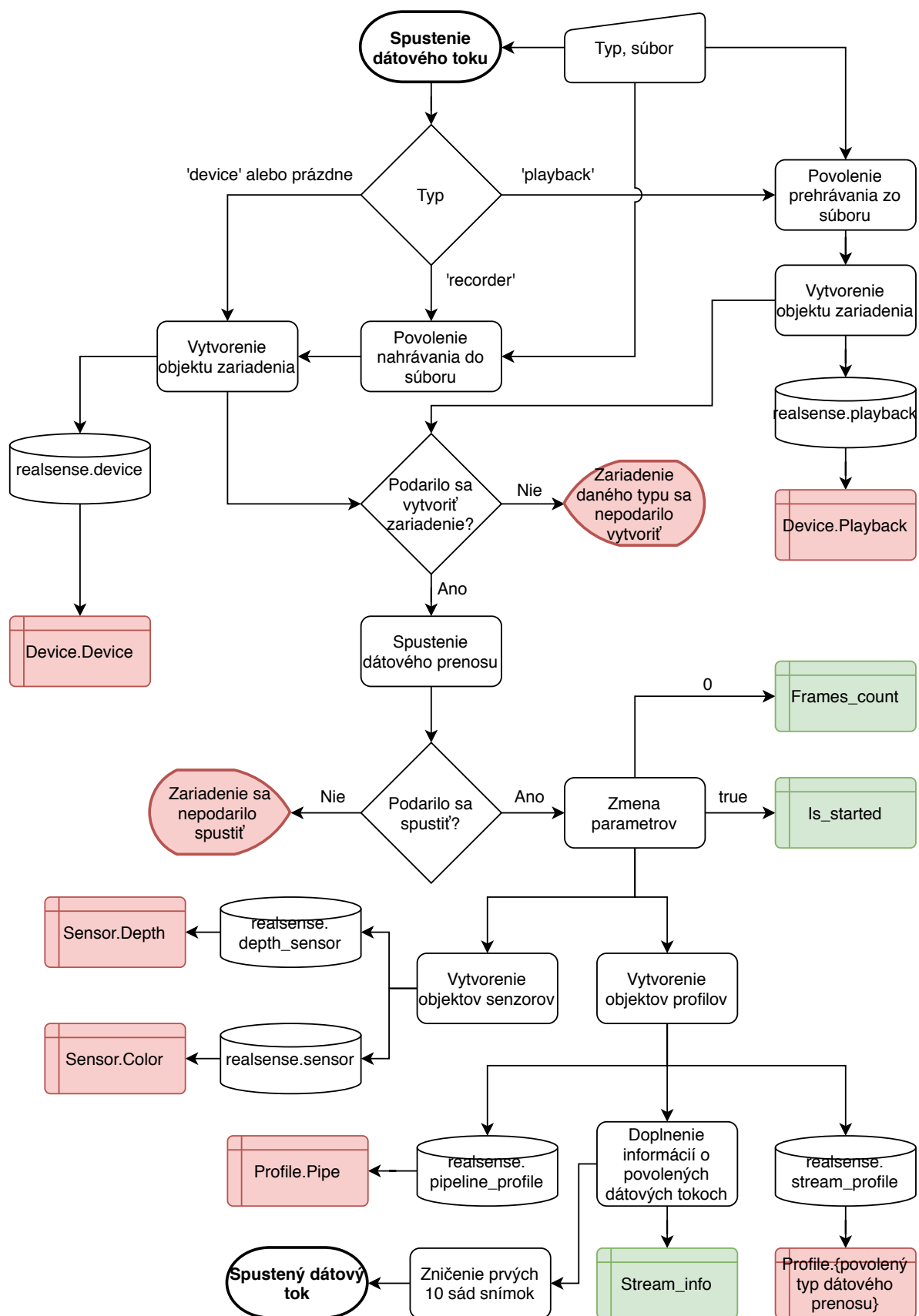
Ďalšou funkcionalitou, ktorú je možné nastaviť pred spustením dátového toku, sú filtre. Použitím metódy *enable\_filter()* je možné filtre inicializovať a metódou *disable\_filter()* zasa odstrániť. Dostupné filtre sú uložené v parametri *Enum.Filters*. Argumentom metódy je rad buniek alebo reťazcov obsahujúci názvy filtrov, napr. `{'colorizer', 'disparity'}` alebo `["colorizer", "disparity"]`. Filtre sú inicializované vo východných nastaveniach, ktoré je možné neskôr podľa potreby zmeniť. Nie je potrebné inicializovať ich pred spustením dátového toku, ich povolenie alebo zakázanie rovnako ako zmena ich nastavení je možná aj po spustení dátového toku.



Obr. 3.3: Diagram popisujúci princíp nastavenia zariadenia pred spustením

### 3 KOMUNIKAČNÝ PROTOKOL

Samotné spustenie dátového toku je komplexný proces, pri ktorom je vytvorené veľké množstvo objektov z podporného balíčka. Dátový tok je možné spustiť použitím metódy *start()*.



Obr. 3.4: Diagram popisujúci princíp metódy *start()* (spustenie dátového toku)



### 3 KOMUNIKAČNÝ PROTOKOL

Princíp tejto metódy popisuje diagram na obrázku 3.4. Metóda môže byť použitá bez argumentov, čoho výsledkom je spustenie dátového toku v režime *device*, teda zariadenia pracujúceho v reálnom čase. Zariadenia RealSense je však možné použiť aj v iných režimoch.

Režim nahrávača je možné spustiť zadaním argumentu *'recorder'* do metódy *start()*. Zároveň je nutné zadať druhý argument obsahujúci cestu a názov súboru, do ktorého sa nahrávka má uložiť. Nahrávky je možné ukladať do súborov s koncovkou *.bag*. Po spustení dátového toku v tomto režime nie je nutné použiť ďalšie funkcie. Vytvorenie objektu *'recorder'* z podporného balíčka v MATLAB nie je možné. Jeho inicializácia spôsobí kritickú chybu, po ktorej je nutné MATLAB znova spustiť. Z tohto dôvodu v RSO tento objekt nie je inicializovaný a nahrávanie je tak možné iba spustiť a zastaviť. Prvky ako pozastavenie a obnovenie nahrávania nie je možné použiť. Do súboru sú ukladané získané sady snímok. Nahrávanie je vypnuté zastavením dátového toku metódou *stop()*. Pri nahrávaní je potrebné dbať na dostatok voľnej pamäte, už pri niekoľkých desiatkach sekúnd môže veľkosť súboru dosiahnuť hodnoty gigabytov.

Ďalším z režimov je režim prehrávača. Režim prehrávača je aktivovaný zadaním argumentu *'playback'* do metódy *start()*. Taktiež musí byť zadaný druhý argument označujúci cestu a názov súboru s príponou *.bag*. Narozdiel od predchádzajúcich dvoch režimov, v tomto prípade nie je potrebné nastavovať dátové toky, toto nastavenie je vyčítané z uloženého súboru. Zariadenie je teda spätne nastavené tak ako bolo nastavené pri nahrávaní daného súboru. Jednotlivé sady snímok sú získavané rovnako ako pri režime snímania v reálnom čase. Prehrávanie je možné ovládať metódami na to určenými, ako je predstavené v jednej z priložených ukážok.

#### 3.2.3 Nastavenie zariadenia

Farebný a hĺbkový senzor spolu s filtermi je možné nastaviť podľa potreby. V RSO je možné ich nastavenie zmeniť tromi spôsobmi.

Prvý zo spôsobov je zmena jednotlivých nastavení vybraného objektu. Pre získanie informácií o nastaveniach daného objektu slúži metóda *get\_all\_options()*. Argumentom metódy je názov objektu, ktorého informácie chce užívateľ získať. Týmto argumentom môže byť *'color'* alebo *'depth'* pre zistenie dostupných nastavení farebného alebo hĺbkového senzoru, prípadne ktoríkoľvek z prvkov parametru *Enum.Filters*. Výstupom z funkcie je rad štruktúr obsahujúci informácie o všetkých dostupných nastaveniach. Príklad vzhľadu tejto štruktúry je na obrázku 3.5, kde je možné vidieť výstup metódy zadanej v tvare *get\_all\_options('depth')*.

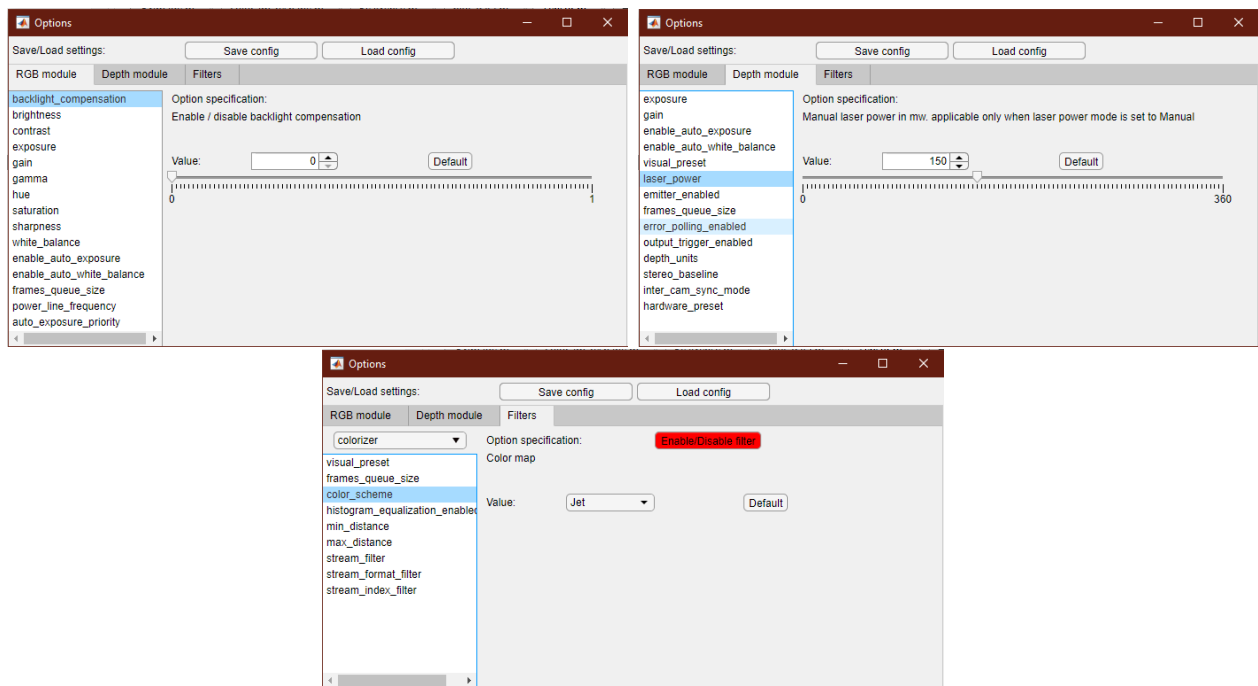
Fields	option	option_spec	min	max	step	def	set	value_spec	value_json
1	"exposure"	'Depth Exposure (usec)'	1	165000	1	33000	33000 []		1x1 struct
2	"gain"	'UVC image gain'	16	248	1	16	16 []		1x1 struct
3	"enable_auto_exposure"	'Enable Auto Exposure'	0	1	1	1	1 []		1x1 struct
4	"enable_auto_white_balance"	'Enable Auto White Balan...	0	1	1	1	1 []		1x1 struct
5	"visual_preset"	'Advanced-Mode Preset'	0	6	1	0	0 {'Custom':"0","Default":"1","Hand":"2","High Accura...		1x1 struct
6	"laser_power"	'Manual laser power in m...	0	360	30	150	150 []		1x1 struct
7	"emitter_enabled"	'Emitter select, 0-disable ...	0	2	1	1	1 {'Off':"0","Laser":"1","Laser Auto":"2"}		1x1 struct
8	"frames_queue_size"	'Max number of frames y...	0	32	1	16	16 []		1x1 struct
9	"asic_temperature"	'Current Asic Temperatur...	-40	125	0	0	35 []		1x1 struct
10	"error_polling_enabled"	'Enable / disable polling ...	0	1	1	0	1 {'Disable error polling':"0","Enable error polling':"1"}		1x1 struct
11	"projector_temperature"	'Current Projector Tempe...	-40	125	0	0	30 []		1x1 struct
12	"output_trigger_enabled"	'Generate trigger from th...	0	1	1	0	0 []		1x1 struct
13	"depth_units"	'Number of meters repres...	1.0000e-06	0.0100	1.0000e-06	0.0010	0.0010 []		1x1 struct
14	"stereo_baseline"	'Distance in mm between...	55.0674	55.0674	0	55.0674	55.0674 []		1x1 struct
15	"inter_cam_sync_mode"	'Inter-camera synchroniz...	0	2	0	1	0 []		1x1 struct
16	"hardware_preset"	'Enable/Disable global ti...	0	1	1	1	1 []		1x1 struct

Obr. 3.5: Všetky dostupné nastavenia a informácie o nich hĺbkového senzoru modelu D415

### 3 KOMUNIKAČNÝ PROTOKOL

Zmenu nastavenia je možné previesť použitím metódy *set\_option()*. Táto metóda potrebuje tri argumenty. Prvým argumentom je názov objektu, ktorého nastavenie sa má zmeniť, napr. *'depth'* pre zmenu nastavenia hĺbkového senzoru. Druhým argumentom je názov nastavenia, ktoré sa má zmeniť, napr. *'visual\_preset'*, a tretím z argumentov je hodnota, na ktorú sa má nastavenie zmeniť. Pokiaľ je zadaná hodnota mimo povolený rozsah alebo nie je násobkom parametra *step*, je nastavená najbližšia možná hodnota daného nastavenia.

Druhým spôsobom je zmena nastavení v GUI slúžiacom práve pre zmenu nastavení zariadenia RealSense. Po spustení dátového toku (vytvorení všetkých potrebných objektov) je možné otvoriť GUI použitím metódy *options()*. Výhodou použitia tohto spôsobu nastavenia zariadenia je možnosť sledovať vplyv jednotlivých nastavení na získané dáta v reálnom čase.



Obr. 3.6: Snímky z GUI slúžiaceho na zmenu nastavení zariadení RealSense

Ako je možné vidieť na obrázku 3.6, nastavenia farebného a hĺbkového senzoru a taktiež nastavenia filtrov sú rozdelené do skupín pre zlepšenie prehľadnosti. V každej skupine je zoznam dostupných nastavení pre daný objekt. Nastavenie je možné zmeniť použitím okienka, do ktorého je možné priamo zapísať hodnotu, alebo použitím slideru. Hodnoty všetkých nastavení sú číselné. Hodnoty niektorých nastavení však majú špecifický význam, v týchto nastaveniach sú okienko a slider nahradené za roletové menu. Pre rýchlu zmenu nastavenia na východziu slúži tlačidlo *Default*. Po kliknutí na toto tlačidlo je hodnota nastavenia zmenená na východziu. V sekcii určenej pre zmenu nastavení filtrov sú dva doplnkové elementy. Keďže je viac druhov filtrov, bolo by zbytočné vytvárať pre každý samostatnú sekciu. Medzi jednotlivými filterami je možné prepínať pomocou roletového menu. Filtre je možné taktiež vytvárať alebo odstraňovať, čo zastáva funkciu inicializácie filtrov spomínanej v kapitole 3.2.2. Po nastavení zariadenia podľa aktuálnych požiadaviek je možné všetky nastavenia uložiť kliknutím na tlačidlo *Save* do konfiguračného textového súboru pre opakované použitie. Konfiguračný súbor je súbor s príponou *.ini*. Dáta v ňom sú uložené do sekcií. V každej sekcii sú kľúče, ktorých názvy tvoria názvy jednotlivých nastavení. Hodnoty jednotlivých kľúčov tvoria hodnoty daných nastavení. Okrem nastavení sú v konfiguračnom súbore uložené informácie o filtroch, ktoré sa majú po načítaní súboru vytvoriť. Ukážkový konfiguračný súbor je priložený v prílohách tejto práce.

### 3 KOMUNIKAČNÝ PROTOKOL

Tretím spôsobom ako zmeniť nastavenia je načítanie konfiguračného súboru vytvoreného vyššie zmieneným spôsobom. Konfiguračný súbor je možné načítať použitím metódy `load_config()`. Potrebným argumentom metódy je cesta a názov súboru, napr. `load_config('ini\default.ini')`

Prácu s konfiguračnými súbormi umožňuje prídavný balíček *INI Config*. [27]

Pre prácu so zariadením je potrebné uviesť niekoľko podstatných nastavení. Pri farebnom senzore sa jedná o nastavenia pre automatickú úpravu kvality snímky, ako je nastavenie `enable_auto_exposure`, teda povolenie automatickej expozície. Povolením automatickej expozície je snímaču umožnené regulovať čas uzávierky, teda dobu, po ktorú sú svetlocitlivé bunky snímača vystavené svetlu. Podobne aj možnosť `enable_auto_white_balance`, teda povolenie automatického vyváženia bielej farby, je užitočné povoliť. Automatická expozícia však môže výsledný obraz taktiež znehodnotiť, pokiaľ je senzor vystavený priamemu svetlu. V týchto prípadoch je vhodné použiť metódu `set_sensor_roi()`, pomocou ktorej je možné určiť oblasť záujmu, na ktorú sa má automatická expozícia zamerať. Posledným zaujímavým nastavením farebného snímača je nastavenie `power_line_frequency`, ktoré pri nastavení na hodnotu 60 Hz dokáže filtrovať šum na snímke spôsobený okolitou elektrickou rozvodnou sieťou.

Dôležitým nastavením hĺbkového senzoru je, okrem vyššie spomenutých nastavení automatickej expozície a automatického vyváženia bielej farby, dôležité nastavenie `visual_preset`, ktoré umožňuje výber spomedzi niekoľkých prednastavených režimov ako je napríklad režim vysokej presnosti alebo veľkej hustoty bodov. Vplyv tejto možnosti na výstupné dáta je možné dobre pozorovať pri práci s mračnom bodov. Veľmi dôležité nastavenie je `depth_units`, v preklade jednotky hĺbky. Východzie nastavenie je na hodnotu 0.001, čo značí, že dáta obsahujúce polohu bodov v priesore majú presnosť 1 mm. Pri takto nastavenej hodnote je maximálna snímateľná vzdialenosť stanovená na približne 65 m, čo je vzhľadom na presnosť stanovenia hĺbky zbytočne veľká vzdialenosť. Zmenou hodnoty jednotiek hĺbky na hodnotu 0.0001 je možné získať dáta s presnosťou na 0.1 mm a zároveň obmedziť maximálnu vzdialenosť na 6.5 m, čo je vhodné pri snímaní na krátku vzdialenosť. Dá sa teda povedať, že závislosť maximálnej vzdialenosti na zvolených jednotkách hĺbky je určená vzťahom 3.1, kde  $d_{max}$  je maximálna snímateľná vzdialenosť v metroch a `depth_units` je hodnota nastavenia. [26]

$$d_{max} = 65000 \cdot depth\_units \quad (3.1)$$

#### 3.2.4 Získavanie a postprocesovanie dát

Zo zariadenia RealSense modelu D415 je možné získať dáta z niekoľkých rozličných snímok. Keďže sa jedná o zariadenie snímajúce hĺbku, najdôležitejšie dáta, ktoré je možné získať, sú dáta hĺbkové. Tieto dáta je možné získať použitím metódy `get_depth_data()`. Pre zlepšenie kvality dát je možné zadať metóde argument obsahujúci názvy filtrov, ktoré majú byť použité pre postprocesovanie hĺbkovej snímky. Hĺbková snímka je automaticky spracovaná filtrom `colorizer`, ktorý každému bodu pridelí farbu pre optické rozlíšenie hĺbky v obraze. Výstupom metódy je tak matica o veľkosti  $M \times 3$ , kde  $M$  je počet pixelov v snímke (všeobecne výška krát šírka snímky) a stĺpce reprezentujú farebný kód daného pixelu.

Dáta z farebného senzoru je možné získať použitím metódy `get_color_data()`. Výstupom metódy je taktiež matica o veľkosti  $M \times 3$ , kde  $M$  značí počet pixelov snímky a stĺpce určujú farebný kód daného pixelu.

### 3 KOMUNIKAČNÝ PROTOKOL

Metódou *get\_infrared\_data()* je možné získať infračervenú snímku. Zadaním argumentu je možné určiť, z ktorej infračervenej kamery majú dáta pochádzať. Výsledná snímka je čierno-biela. Výstupom metódy sú dáta vo forme vektora s M riadkami reprezentujúcimi počet pixelov v snímke.



Obr. 3.7: Diagram popisujúci princíp získavania dát zo zariadenia RealSense

### 3 KOMUNIKAČNÝ PROTOKOL

Vyššie popísané získané dáta sú však stále vo formáte vhodnom pre prezentovanie obrazového výstupu. Použitím nasledujúcich dvoch metód je možné získať namerané hodnoty vzdialenosti. Pre získanie nameraných hodnôt vzdialenosti pre menší počet bodov je vhodné použiť metódu *get\_point\_distance()*. Zadaním súradníc bodov, ktorých vzdialenosť má byť získaná, je možné rýchlo získať ich vzdialenosť. Výhodou funkcie je zadávanie polohy bodov ako súradníc X a Y z obrazu, čo umožňuje implementáciu funkcie v kombinácii s detektormi objektov v obraze.



Obr. 3.8: Diagram popisujúci princíp získavania dát pre mračno bodov

### 3 KOMUNIKAČNÝ PROTOKOL

Na obrázku 3.8 je diagram popisujúci princíp získavania dát pre mračno bodov. Metódou *get\_pointcloud()* sú získavané dáta potrebné pre vytvorenie mračna bodov. Tento dátový typ sa skladá z polohy bodov v priestore (súradnice X, Y a Z) a informácie o farbe každého bodu. Pre získanie správnych farebných údajov je potrebné zarovnať hĺbkovú a farebnú snímku, k čomu je využívaný *align* filter, ktorý je metódou automaticky vytvorený pokiaľ nebol vytvorený pri inicializácii dátového toku. Výstupom funkcie je matica Mx3 obsahujúca informácie o polohe bodov a druhá matica rovnakej veľkosti obsahujúca informácie o farbe daných bodov. Informácie o polohe bodov sú získavané použitím *pointcloud* filtra.

Výstupmi vyššie zmienených metód sú dáta vo formáte *double*, prípadne *uint8* (údaje o farbe). Pokiaľ je snímanie a vykresľovanie dát príliš pomalé, je možné využiť akceleráciu spracovania dát pomocou grafickej karty pridaním argumentu '*gpuarray*' do metód získavajúcich dáta. Tým je formát získaných dát pretvorený na formát *gpuArray*, ktorý umožňuje spracovávať dáta na grafickej karte počítača.

Postprocesovanie dát spočíva v postupnom spracovávaní získanej hĺbkovej snímky filtermi. Pri filtrovaní je dôležité dodržiavať istú postupnosť a náväznosť jednotlivých filtrov na seba. Proces filtrovania hĺbkových snímkov je znázornený v diagrame na obrázku 3.9.

Pokiaľ je hodnota parametru *Align* nastavená na *true*, užívateľom alebo automaticky pri použití metódy *get\_pointcloud()*, získaná sada snímkov je spracovaná *align* filtrom, ktorý zarovná hĺbkovú a farebnú snímku tak, aby každý pixel na hĺbkovej snímke odpovedal tomu istému pixelu na farebnej snímke. Takto zarovnaná sada snímkov pokračuje na ďalšie spracovanie. [26]

Ďalším v poradí je decimálny filter, ktorého úlohou je zmeniť rozmery snímky podľa nastaveného stupňa filtrovania, napr. pokiaľ je stupeň filtrovania nastavený na hodnotu 2, rozmery snímky sú dvojnásobne zmenšené. Takéto zpracovanie snímky je prospešné najmä pri získavaní dát pre mračno bodov. Pri zmene rozmerov snímky sú hĺbkové hodnoty nových pixelov vytvorené ako nenulový priemer pôvodných pixelov, čím sa dáta vyhladzujú. Taktiež pri spracovaní dát z mračna bodov je potrebné zredukovať počet bodov pre urýchlenie procesu. V porovnaní s redukciou bodov použitím funkcie *pcdownsample* v programe MATLAB, je použitie decimálneho filtra oveľa rýchlejšie. [26]

Po zmene rozmerov hĺbkovej snímky je snímka prevedená na pôvodnú snímku rozdielov bodov. Prevod hĺbkovej snímky naspäť na snímku rozdielov polôh bodov má za následok zmenšenie chyby merania a tak zlepšenie kvality výstupných hĺbkových dát. Zároveň je výrobcom odporúčané používanie nasledujúcich filtrov práve na snímke rozdielov polôh. [26]

Priestorový (*spatial*) filter má za úlohu vyhladiť šum hĺbkových hodnôt a zároveň vyhladiť roviny. Úroveň filtrovania je nastavená pomocou alfa a delta parametrov v nastaveniach. Nesprávnym nastavením tohto filtra sú znehodnotené údaje o tvare a rozmeroch objektov, preto je odporúčané nastavovať parametre v závislosti na následnom použití výsledných dát. Príliš agresívne filtrovanie má síce za následok vyhladenie rovín ako napríklad steny, ale zároveň spôsobuje príliš veľkú deformáciu objektov iných tvarov. [26]

Dočasný (*temporal*) filter je užitočný postproces pokiaľ sa snímaná scéna rýchlo nemení. Tento filter priemeruje hĺbkové hodnoty z aktuálnej snímky s nenulovými hĺbkovými hodnotami z predchádzajúcich snímkov. Tým sú zosnímané hodnoty viac vyhladené. Ďalšou užitočnou funkcionalitou tohto filtra je dopĺňanie hodnôt do medzier v hĺbkových dátach aktuálnej snímky podľa hodnôt z predchádzajúcich snímkov. Túto funkciu je možné nastaviť na rôzne stupne dopĺňania hodnôt, najužitočnejšie sú však možnosti dopĺňovania bodov, ktoré boli validné v dvoch alebo troch zo štyroch predchádzajúcich snímkov. [26]



Obr. 3.9: Diagram znázorňujúci proces filtrovania hĺbkovej snímky

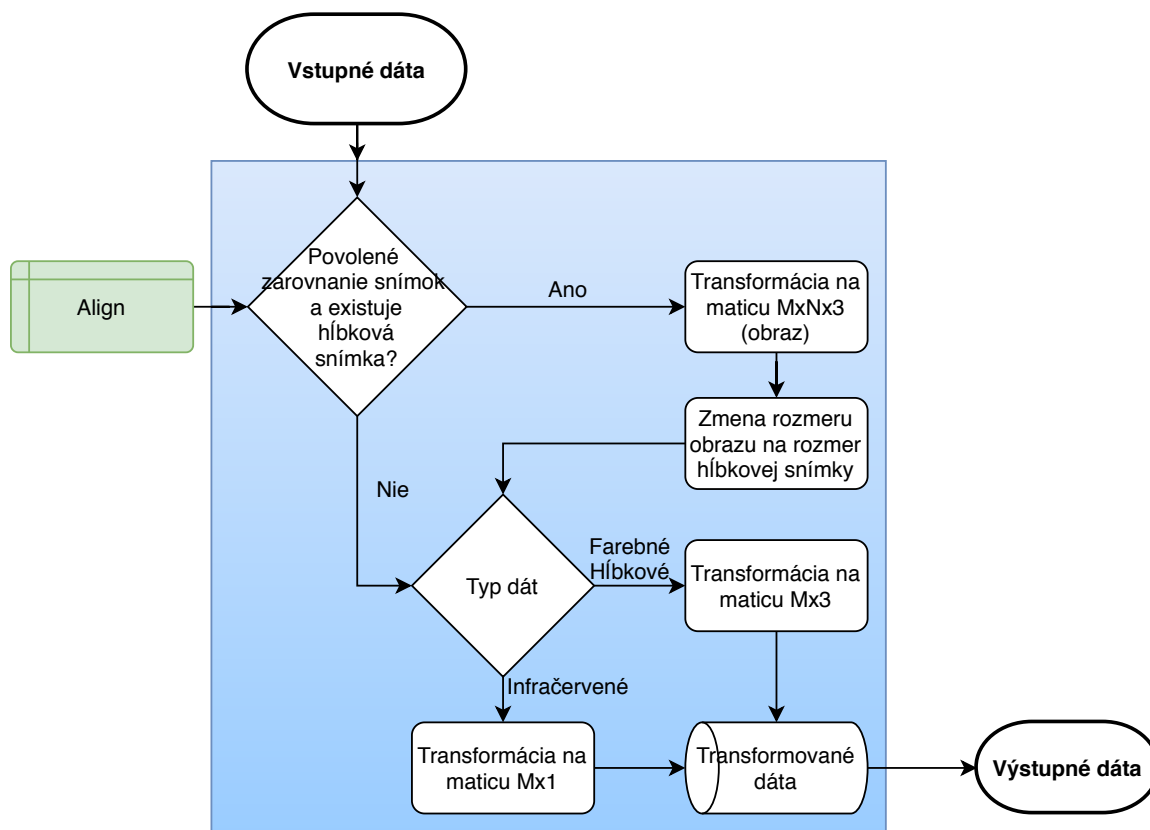
Niektoré aplikácie sú citlivé na chýbajúce hĺbkové údaje. Vtedy je dobré vyplniť chýbajúce miesta v snímke odhadovanou hodnotou hĺbky. K tomu slúži filter doplňujúci diery. Úlohou tohto filtra, ako už jeho názov napovedá, je doplniť chýbajúce hĺbkové údaje hodnotami dopočítavanými z hĺbkových údajov okolitých bodov. Pri aplikáciách, v ktorých je potrebné presné rozoznávanie objektov je odporúčané tento filter nepoužívať. [26]

Po prefiltrovaní snímky rozdielu polôh bodov je potrebné previesť snímku naspäť na hĺbkovú.

### 3 KOMUNIKAČNÝ PROTOKOL

Pokiaľ majú byť získané hĺbkové dáta vo forme obrazu (je požadovaný výstup z metódy *get\_depth\_data()*) je hĺbková snímka spracovaná filtrom *colorizer*, ktorý pridelí jednotlivým pixelom farbu v závislosti na ich vzdialenosti. Výstupom z tohto filtra je vektor dát, ktorý je pre účely ďalšieho použitia transformovaný na maticu  $M \times 3$ .

Transformácia vektoru na maticu je automatická a využíva sa aj pri získavaní dát zo zvyšných dátových tokov. Účelom transformácie dát je ich prevedenie na jednotnú formu, a to na maticu s počtom riadkov odpovedajúcim počtu pixelov v snímke. Stĺpce matice charakterizujú farbu jednotlivých pixelov. Transformácia dát zároveň zastáva funkciu decimálneho filtra pre filtrovanie farebnej a infračervenej snímky. V prípade, že je parameter *Align* nastavený na logickú pravdivú hodnotu *true* a je vytvorená hĺbková snímka, sú dáta získané z farebného a infračerveného dátového toku transformované na obraz ( $M \times N \times 3$  alebo  $M \times N \times 1$ ). Rozmery obrazu sú následne zmenené na rozmery hĺbkovej snímky a naspäť transformované na maticu  $M \times 3$  alebo  $M \times 1$ . Proces transformácie je dostatočne rýchly na to, aby neovplyvnil rýchlosť získavania dát. V prípade potreby je možné využiť akceleráciu procesu grafickou kartou pridaním argumentu *gpuarray* do metódy, ktorá tieto dáta získava, ako bolo spomenuté vyššie. Proces transformovania dát je popísaný diagramom na obrázku 3.10.



Obr. 3.10: Diagram popisujúci princíp transformácie dát



### 3.3 Testovanie rýchlosti snímania a presnosti merania

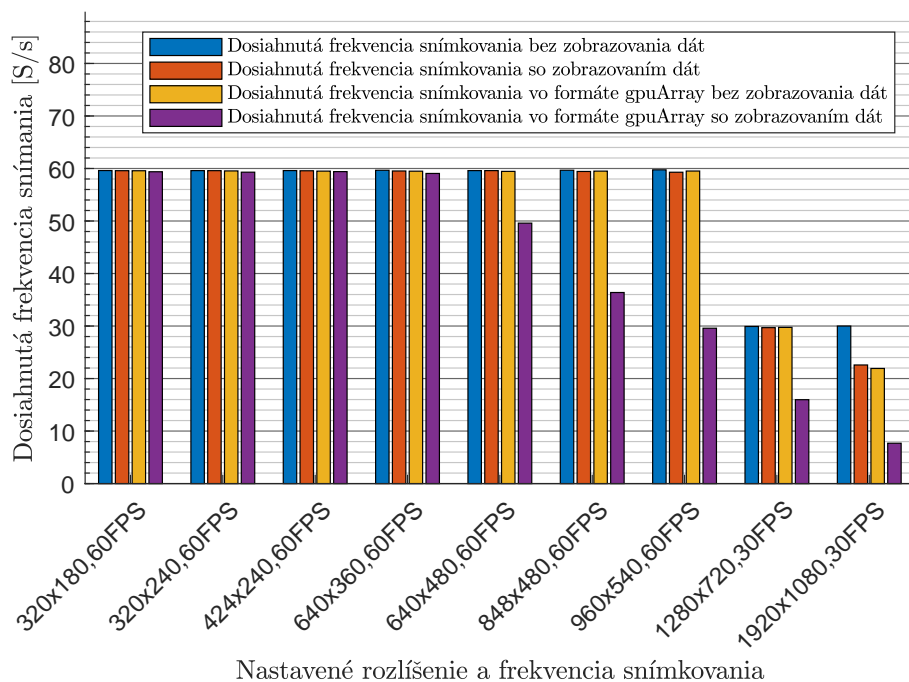
Komunikačný protokol bol podrobený testu rýchlosti snímania pri rôznych nastaveniach rozlíšenia snímok. Test spočíval v určení rýchlosti získavania dát z jednotlivých snímok bez ich vykresľovania a následne s vykresľovaním. Súčasťou testu je aj porovnanie získavania, spracovania a vykreslenia dát využitím grafickej karty.

Testovanie rýchlosti získavania dát z farebného senzoru:

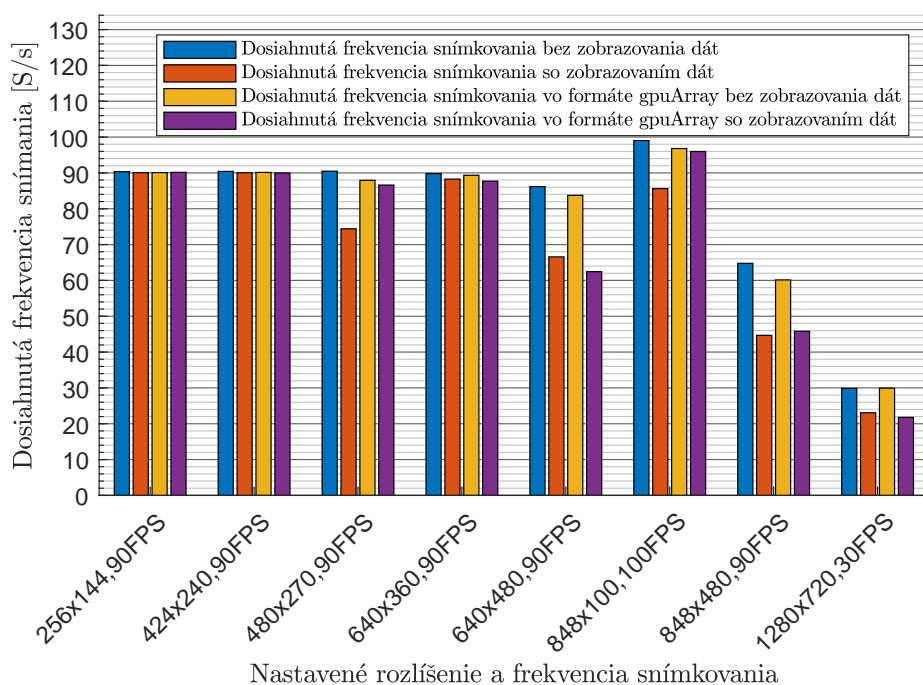
1. Pred spustením merania je vytvorený prehrávač, pomocou ktorého sú zosnímané dáta vykresľované. Pre vykresľovanie dát je použitý prehrávač *VideoPlayer* z knižnice *Computer Vision*. Výhodou tohto prehrávača je oproti funkcii *imshow()* rýchlosť, ktorou je schopný dáta vykresľovať.
2. Je nastavené najnižšie podporované rozlíšenie s najvyššou možnou frekvenciou snímokovania a je spustený dátový tok.
3. *For* cyklom je vytvorených sto snímok, z ktorých sú získané dáta. Pri každej iterácii cyklu sú z vytvorenej snímky získané dáta a funkciami *tic* a *toc* je zaznamenaný čas potrebný na ich získanie.
4. Po ukončení cyklu je vypočítaný priemerný čas získavania dát zo snímky a vzťahom  $f = 1/T$ , kde  $T$  je priemerný čas získavania snímky a dát, je určená frekvencia snímokovania.
5. Body 3 až 4 sa opakujú, avšak dáta sú po získaní aj vykreslené, je určený priemerný čas získavania a vykresľovania a frekvencia snímokovania.
6. Body 3 až 5 sa opakujú, pričom po vytvorení snímky sú dáta získané, spracované a vykreslené použitím grafickej karty a je určená frekvencia snímokovania takto spracovaných dát.
7. Dátový prenos je ukončený, je zvýšené rozlíšenie a nastavená najvyššia možná rýchlosť snímokovania, dátový prenos je opäť spustený a je prevedené meranie pri novom rozlíšení.

Tento postup sa opakuje až pokiaľ nie je dosiahnuté maximálne rozlíšenie senzoru. Testovanie rýchlosti získavania hĺbkových a infračervených dát pracuje na rovnakom princípe ako meranie rýchlosti získavania farebných dát. Pri testovaní rýchlosti získavania farebných a hĺbkových dát je rozdiel v tom, že sú v jednotlivých cykloch naraz získavané, spracovávané a vykresľované ako farebné, tak aj hĺbkové dáta. Testovanie rýchlosti vytvárania mračna bodov rozširuje test rýchlosti získavania farebných a hĺbkových dát o vytvorenie mračna bodov zo získaných dát a jeho vykreslenie. Mračno bodov je vykresľované v objekte *pcplayer*, ktorý je určený práve na zobrazovanie mračien bodov. Pri testovaní rýchlosti získavania farebných a hĺbkových dát súčasne a pri získavaní mračna bodov sú využité iba také rozlíšenia snímok, ktoré majú tieto dátové toky spoločné.

Test je prevedený skriptom *DATesting.m* priloženým k práci. Po ukončení merania boli získané dáta uložené do súboru *DATesting.mat*, ktorý je taktiež súčasťou príloh.

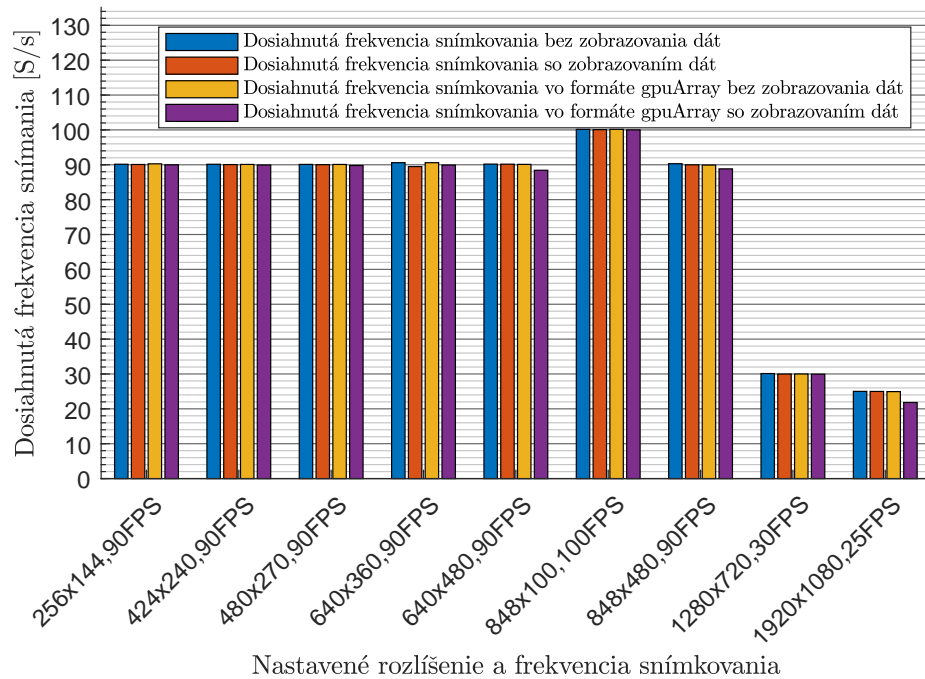


Obr. 3.11: Graf zobrazujúci rýchlosť získavania, spracovania a vykreslenia farebných dát

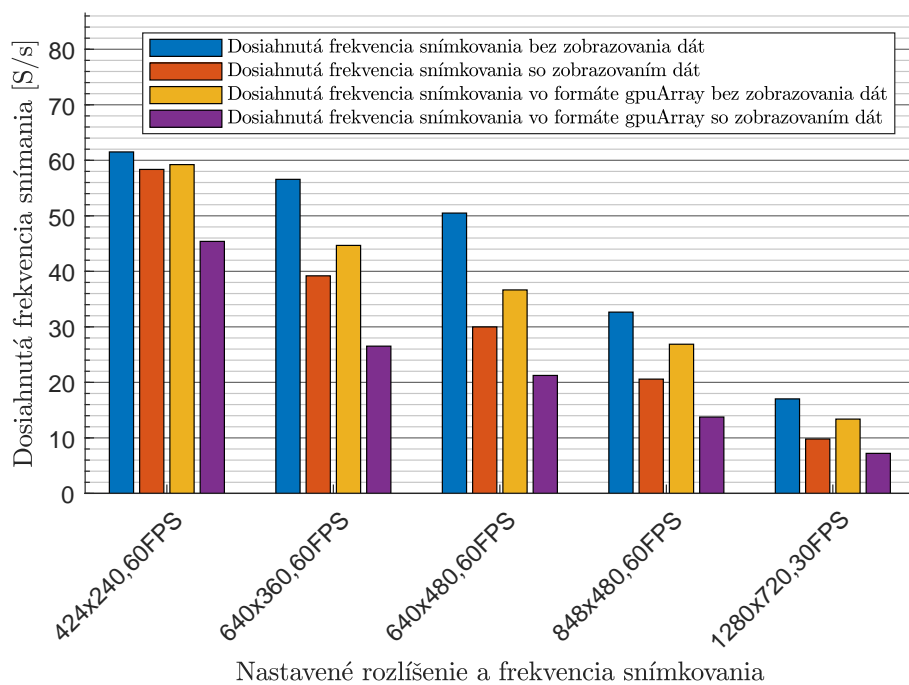


Obr. 3.12: Graf zobrazujúci rýchlosť získavania, spracovania a vykreslenia hĺbkových dát

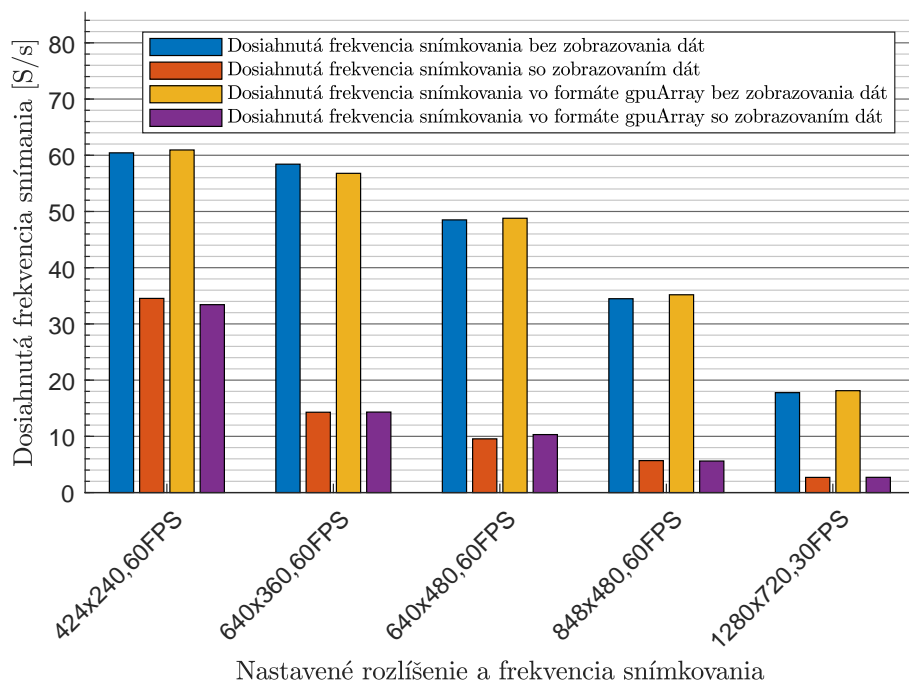
### 3 KOMUNIKAČNÝ PROTOKOL



Obr. 3.13: Graf zobrazujúci rýchlosť získavania, spracovania a vykreslenia infračervených dát



Obr. 3.14: Graf zobrazujúci rýchlosť získavania, spracovania a vykreslenia farebných a hĺbkových dát



Obr. 3.15: Graf zobrazujúci rýchlosť získavania, spracovania a vykreslenia mračna bodov

V grafoch 3.11 až 3.13 je možné vidieť, že pri použití RSO pre získavanie dát iba z jedného dátového toku je spracovanie a vykresľovanie dát dostatočne rýchle. Pri nižších rozlíšeniach je frekvencia snímkovania pri získavaní a vykresľovaní dát rovnaká ako nastavená frekvencia snímkovania. Pri vyššom rozlíšení je vidieť menšie spomalenie rýchlosti vykresľovania dát, toto spomalenie je však zanedbateľné.

Pri získavaní dát súčasne z viacerých dátových tokov (graf 3.14) je s rastúcim rozlíšením rýchlosť získavania a vykresľovania dát znateľne nižšia než je nastavená frekvencia snímkovania. To je spôsobené sériovým radením procesov, ktoré dané dáta vytvárajú. Pri bližšom preskúmaní grafov 3.14 a 3.15 je možné si všimnúť, že rýchlosť vytvárania mračien bodov skoro je totožná s rýchlosťou súčasného získavania farebných a hĺbkových dát. Rozdiel medzi týmito dvoma procesmi je ich výstup. Pri súčasnom získavaní dát sú výstupom hĺbkové a farebné dáta vo forme obrázku (resp. matíc, ktoré je možné pretvárať na formát obrázku  $M \times N \times 3$ ), pričom výstupom pri získavaní mračna bodov je objekt *pointCloud*, ktorý obsahuje údaje o polohe a farbe jednotlivých pixelov získaných zo snímok. Objekt *pointCloud* je súčasťou knižnice *Computer Vision*, ktorá obsahuje množstvo funkcií užitočných pri spracovávaní mračien bodov. Tieto funkcie budú bližšie predstavené v nasledujúcej kapitole. Mračná bodov sú zobrazované pomocou prehrávača *pcplayer*. Ich zobrazovanie je však pomalý proces, ako je možné vidieť na grafe 3.15. Pri práci s mračnami bodov je odporúčané použiť decimálny filter so stupňom filtrovania 2 až 3, ktorý okrem zlepšenia kvality mračna znižuje počet bodov, čo vo výsledku umožňuje rýchlejšie zobrazovanie mračien bodov.

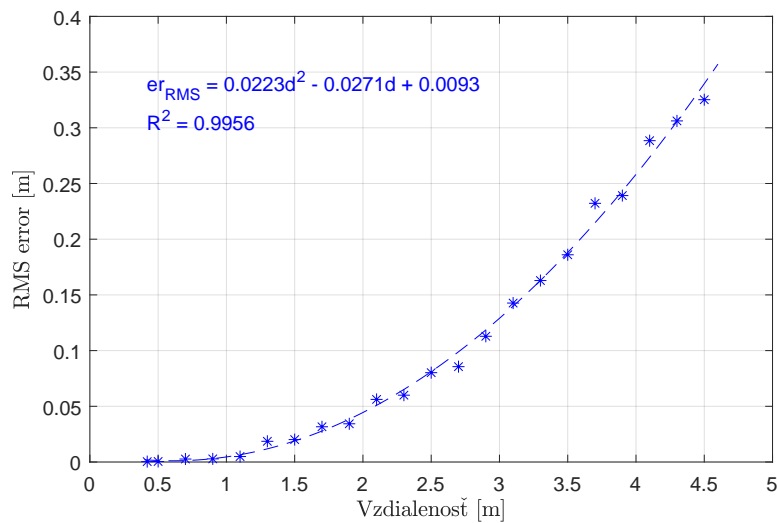
V grafoch je možné vidieť, že spracovávanie získaných dát na grafickej karte za účelom zobrazovania dát má skôr negatívny účinok, čo je spôsobené prevodom dát na *gpuArray* a spätným prevodom na typ *double* alebo *uint8*. Spracovanie dát grafickou kartou môže byť prospešné obzvlášť pri ďalšom postprocesovaní dát vo forme obrázkov. Objekt *pointCloud* nepodporuje prácu s formátom *gpuArray*, preto je pre ďalšie spracovanie mračien bodov nutné využívať

### 3 KOMUNIKAČNÝ PROTOKOL

výpočet pomocou procesoru.

Rýchlosti získavania, spracovávanía a zobrazovania dát sa môžu líšiť v závislosti na výkone procesora a grafickej karty počítača.

Ďalším testom, ktorému boli zariadenie RealSense D415 a RSO podrobené je test presnosti merania vzdialenosti. Účelom tohto testu nie je vytvoriť presnú charakteristiku závislosti chyby merania na vzdialenosti, ale potvrdiť, že je závislosť chyby merania na vzdialenosti kvadratická. Krivka, ktorú je možné vidieť na grafe 3.16 môže mať veľa tvarov. Stúpanie krivky je ovplyvnené faktormi ako napríklad okolité osvetlenie, veľkosť, farba, tvar a štruktúra povrchu snímaného objektu, alebo nastavenie senzoru a filtrov.



Obr. 3.16: Graf závislosti chyby merania na vzdialenosti objektu od kamery

V tomto prípade bola miestnosť, v ktorej bol test vykonaný slabo osvetlená chodba bytu a ako objekt, ktorého vzdialenosť bola meraná, je použitá krabička od zariadenia RealSense, ktorá je hladká a z veľkej časti čierna. Tieto podmienky sa odzrkadľujú na prudkom stúpaní krivky.

Zariadenie RealSense malo pri testovaní povolený farebný a hĺbkový dátový tok, oba nastavené na rozlíšenie 1280x720 pixelov s frekvenciou snímkovania 30 S/s. Hĺbkové snímky boli postprocesované decimálnym filtrom, následne prevedené na rozdiel polôh bodov, prefiltrované priestorovým a dočasným filtrom a prevedené naspäť na hĺbku (viď kapitola 3.2.4). Všetky parametre filtrov boli nastavené na východziu hodnotu a stupeň filtrovania decimálneho filtra bol nastavený na hodnotu 1, čiže žiadne filtrovanie. Jednotky hĺbkového senzoru (nastavenie *depth\_units*) boli zmenené na hodnotu 0.0001, čo odpovedá výstupným dátam získaným s presnosťou na 0.1 mm a zároveň obmedzuje maximálnu merateľnú vzdialenosť na približne 6.5 m (viď kapitola 3.2.3). Pred kameru bol rozvinutý meter, po ktorom bol v priebehu merania presúvaný snímaný objekt. Kamera bola nastavená tak, aby nameraná hodnota v okamihu, kedy je objekt vo vzdialenosti 0.5 m, bola čo najpresnejšia. Po nastavení meradla bolo objektom postupne posúvané od vzdialenosti 0.42 m (približne najmensej merateľnej vzdialenosti) až po vzdialenosť 4.5 m. Po nastavení meradla bolo prevedené samotné meranie s objektom vo vzdialenosti 0.42 m, následne 0.5 m, a ďalej s krokom 0.2 m až po vzdialenosť 4.5 m. Pri každom meraní bolo vytvorených sto snímok, zo všetkých snímok bola získaná vzdialenosť pixelu v strede obrazu. Bod, ktorý pixel predstavoval, bol bodom snímaného objektu. Po skončení merania boli namerané dáta vyhodnotené a bol vytvorený graf závislosti chyby merania na

### 3 KOMUNIKAČNÝ PROTOKOL

vzdialenosti (obrázok 3.16). Chyby kvadratického priemeru (*RMS error*) boli vypočítané z dát získaných pri jednotlivých meraniach ako chyba kvadratického priemeru nenulových hodnôt daného merania pomocou vzťahu 3.2. [28]

$$er_{RMS} = \sqrt{\frac{\sum_{i=1}^N (\hat{d} - d_i)^2}{N}} \quad (3.2)$$

Vo vzťahu 3.2 je  $\hat{d}$  vzdialenosť zmeraná pomocou metra (predpokladaná vzdialenosť objektu od kamery),  $N$  je počet nenulových nameraných hodnôt v danom meraní a  $d_i$  sú jednotlivé nenulové hodnoty z daného merania. Takto získané chyby merania boli nakoniec preložené regresnou krivkou (polynómom druhého rádu).



Obr. 3.17: Snímka zo zariadenia RealSense D415 vytvorená pri testovaní presnosti merania

## 4 Spracovanie dát zo zariadenia

### RealSense

Po získaní dát z kamery je potrebné ich následne spracovať. Za účelom spracovania ako obrazových dát (farebné, hĺbkové a infračervené) a mračien bodov bol vytvorený objekt *RSImage* (RSIO). Pri inicializácii RSIO je možné vytvoriť objekty prehrávačov typu *vision.VideoPlayer* pre zobrazovanie obrazových typov dát a *pcplayer* pre zobrazovanie mračien bodov, ktoré následne môžu byť použité pre zobrazovanie spracovaných dát.

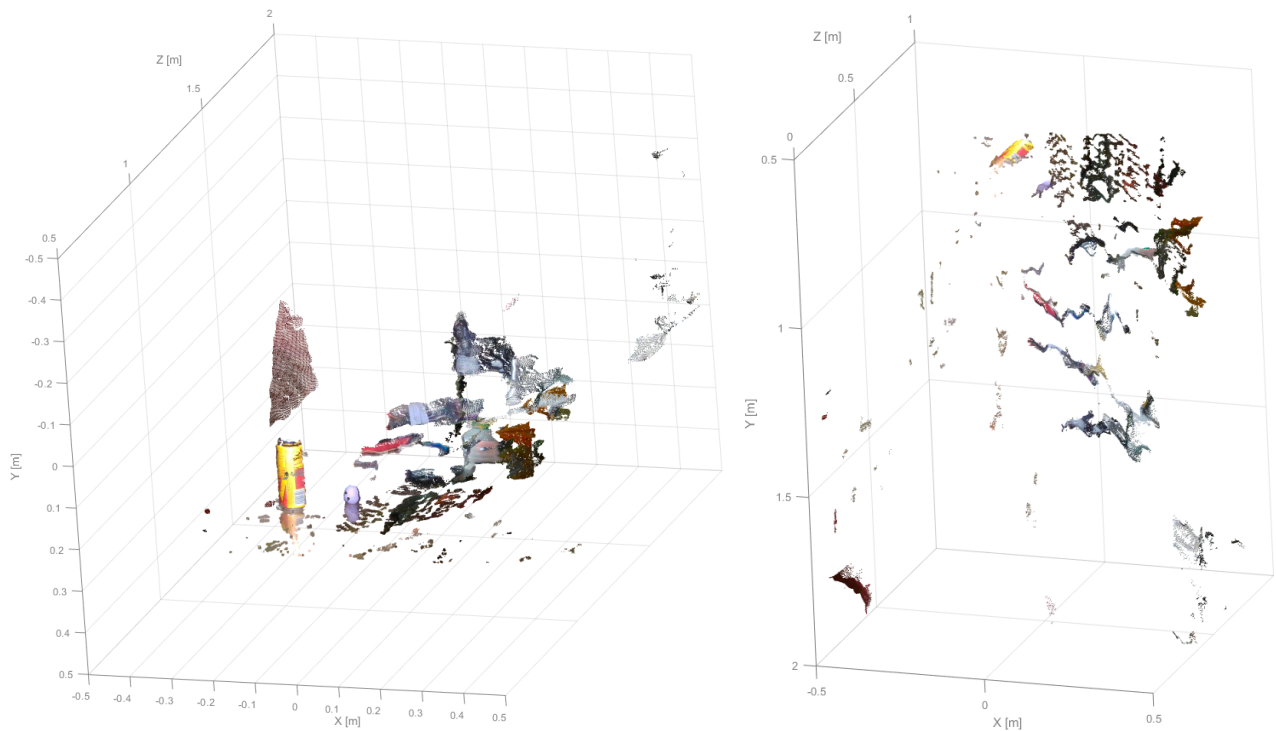
Po použití metódy RSO *get\_pointcloud()* sú získané všetky potrebné dáta pre vytvorenie mračna bodov a jeho ďalšie spracovanie. Vytvorenie mračna bodov vo formáte *pointCloud* (PC) je pre ďalšiu prácu s dátami dôležité, keďže ďalšie metódy RSO pracujú s mračnom bodov práve v tomto formáte. PC je výstupom metódy RSIO *to\_pointcloud()*. Argumentmi metódy sú výstupy metódy RSO *get\_pointcloud()*, čiže priestorové súradnice jednotlivých bodov, ich farba a rozlíšenie snímky. Pokiaľ je medzi argumentmi metódy rozlíšenie pôvodnej snímky, sú tieto hodnoty uložené ako parameter objektu *Pointcloud\_resolution*. Definovanie rozmeru snímky, z ktorej bolo mračno bodov vytvorené, je dôležité. Pri spracovaní mračna metódami RSIO je totiž stále zachovaný počet bodov, z ktorého bolo mračno pôvodne vytvorené, čo umožňuje prevod spracovaného a rozsegmentovaného PC na podobu obrázku. S obrázkom je možné následne pracovať, využiť detekciu zo spracovaného PC napríklad pre urýchlenie práce detektora alebo klasifikátora objektov. Súradnice bodov v takto vytvorených PC sú ukladané v metroch a označujú vzdialenosť jednotlivých bodov od zariadenia, ktoré ich nasnívalo, v tomto prípade kamery RealSense.

Metódy RSIO sú z veľkej časti tvorené funkciami z balíka *ComputerVision*. Jedná sa o funkcie umožňujúce spracovávať dáta uložené v objekte typu *pointCloud*. Samotné používanie funkcií nevedlo k požadovaným výsledkom a tak sú implementované do jednotlivých metód RSIO, v ktorých spolu s ďalšími funkciami a inými prvkami tvoria komplexnejšie algoritmy pre spracovanie mračien bodov.

Počiatok súradnicového systému je možné v PC presúvať a takisto meniť orientáciu jednotlivých osí transformovaním. Transformovanie umožňuje metóda *transform\_pc()*, ktorej argumentmi sú okrem samotného PC, ktoré má byť transformované, aj dva trojprvkové vektory. Jeden z vektorov určuje transláciu počiatku súradnicového systému, druhý zasa jeho rotáciu okolo jednotlivých osí. Po zadaní týchto argumentov do metódy je vytvorená transformačná matica, ktorou je následne PC transformovaný. Transformovanie PC má zmysel najmä vtedy, keď je známa poloha a natočenie senzoru RealSense. Transformovaný PC na obrázku 4.1 vpravo bol transformovaný otočením o 90° okolo osi X a následne bol stred súradnicového systému posunutý o 0.5 m pozdĺž osi Z.

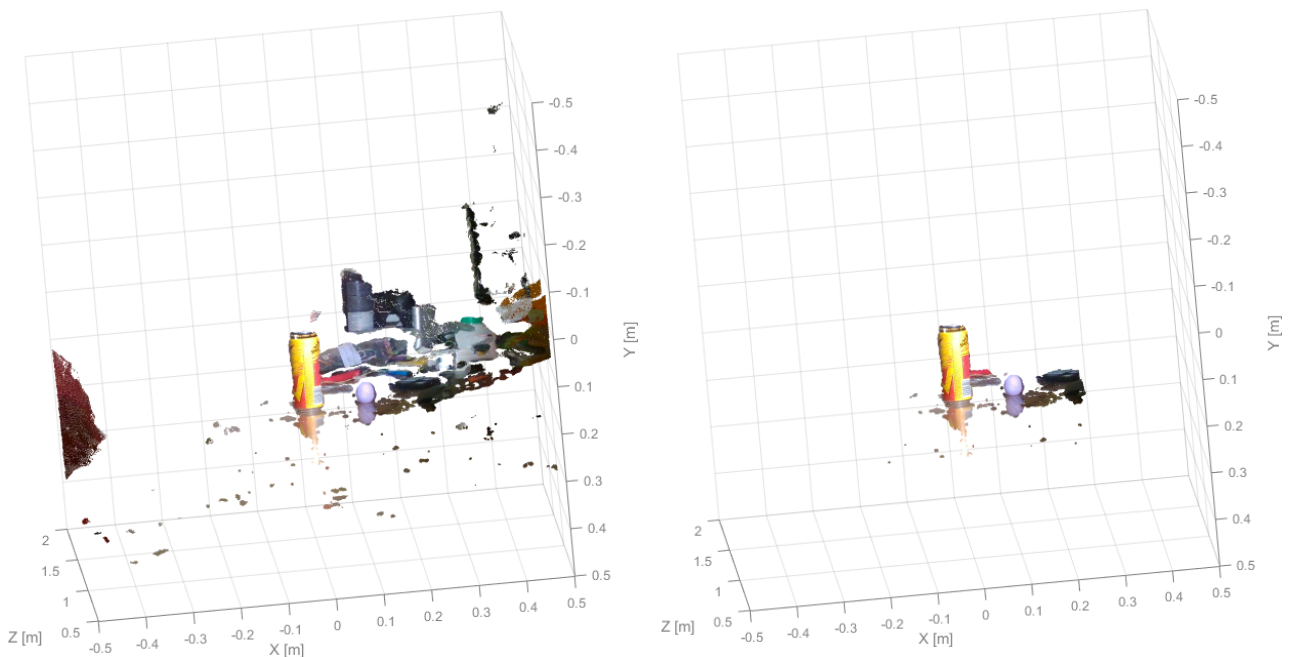


## 4 SPRACOVANIE DÁT ZO ZARIADENIA REALSENSE



Obr. 4.1: Transformácia PC metódou *transform\_pc()*. Vľavo pôvodný PC, vpravo transformovaný PC

Ďalším spôsobom ako upraviť PC je vymazanie bodov nachádzajúcich sa mimo oblasti záujmu. Vymazanie bodov nachádzajúcich sa mimo oblasť záujmu dokáže značne urýchliť ďalšie spracovanie dát. Body ležiace mimo oblasť záujmu umožňuje vymazať metóda *delete\_invalid\_points()*. Oblasť záujmu je matica o veľkosti 3x2, kde riadky matice predstavujú osi X, Y a Z a stĺpce predstavujú minimálnu a maximálnu hranicu, v ktorej sa body musia nachádzať.

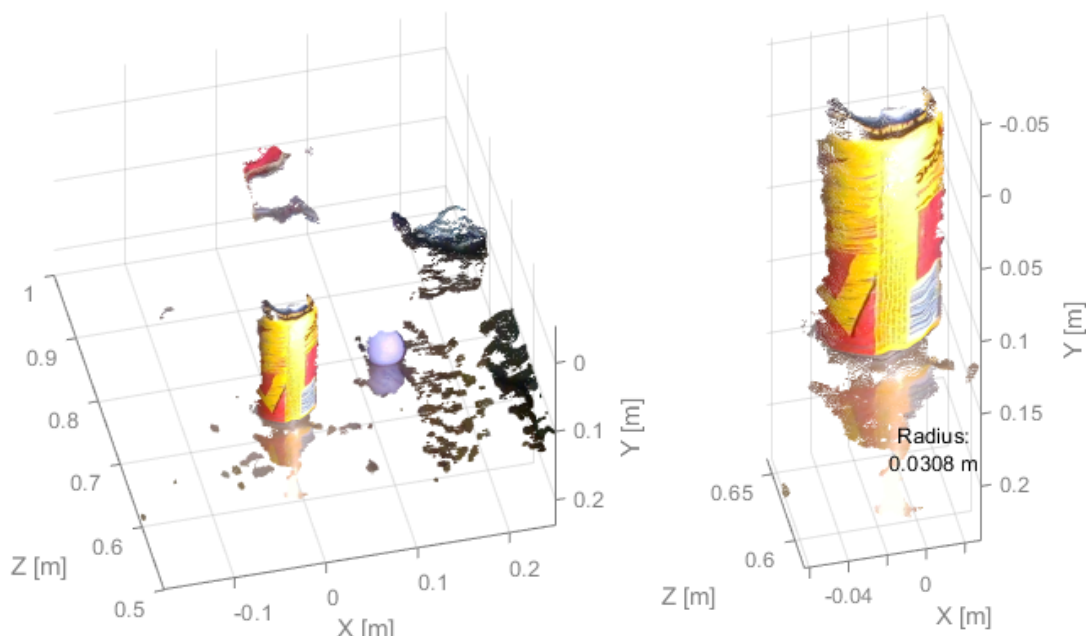


Obr. 4.2: Odstránenie bodov mimo oblasť záujmu. Vľavo pôvodný PC, vpravo PC redukovaný na oblasť záujmu



Výsledok použitia tejto metódy je možné vidieť na obrázku 4.2, kde boli ponechané iba body, ktoré ležali v oblasti záujmu definovanom hranicami  $X \in \langle -0.25, 0.25 \rangle$ ,  $Y \in \langle -0.25, 0.25 \rangle$  a  $Z \in \langle 0.5, 1 \rangle$ .

Samotné zisťovanie tvarov snímaných objektov umožňuje metóda *fit\_shape()*, ktorá v zadanom PC vyhladá body tvoriace sieť, ktorá najviac pripomína hľadaný tvar. Táto metóda využíva funkcie *pcfitplane()*, *pcfitsphere()* a *pcfitcylinder()* z balíka *ComputerVision*. Pri použití metódy je argumentom možné definovať, aký tvar má v PC hľadať. Pre upresnenie hľadania sú metóde zadávané ďalšie argumenty určujúce: v prípade hľadania gule alebo valca limitné hodnoty rádiusu nájdeného objektu, v prípade roviny maximálnu vzdialenosť medzi bodmi a rovinou. Pokiaľ je hľadaným tvarom rovina, je možné určiť smer normálového vektora roviny a jeho maximálnu odchýlku od neho, v prípade hľadania valca je zasa možné určiť smer jeho osi rotácie a maximálnu jeho uhlovú odchýlku. Posledným argumentom zadávaným metóde je hodnota označujúca minimálnu istotu nájdenej tvarovej zhody. Hodnota istoty môže byť akékoľvek číslo medzi nulou a sto, základne je jej hodnota 99. Pokiaľ je požadovaná vyššia presnosť hľadania tvaru, je nutné, aby toto číslo bolo vyššie. Pri detekcii tvaru v PC je vytvorený model daného tvaru a určená odchýlka priemerov vzdialeností bodov, ktoré sú súčasťou detekcie od modelu. Vo verejných parametroch RSIO je parameter s názvom *Pc\_fit\_mean\_error\_threshold*. Jedná sa o hodnotu maximálnej dovolenej priemernej odchýlky vzdialenosti bodov označených ako súčasť detekcie od vytvoreného modelu. Pokiaľ je po prevedení detekcie vypočítaná odchýlka väčšia než maximálna dovolená odchýlka, je vytvorený model považovaný za chybný a následne odstránený. Na obrázku 4.3 je možné vidieť výsledok použitia metódy *fit\_shape()* pri hľadaní valca (plechovky) v PC. Pri hľadaní bola zadaná predpokladaná orientácia osi rotácie s maximálnou uhlovou odchýlkou  $2^\circ$  a hodnotou istoty 99.5. Plechovka bola úspešne nájdená a jej polomer bol stanovený na hodnotu  $R = 0.0308$  m, pričom skutočný polomer plechovky je 0.0325 m.

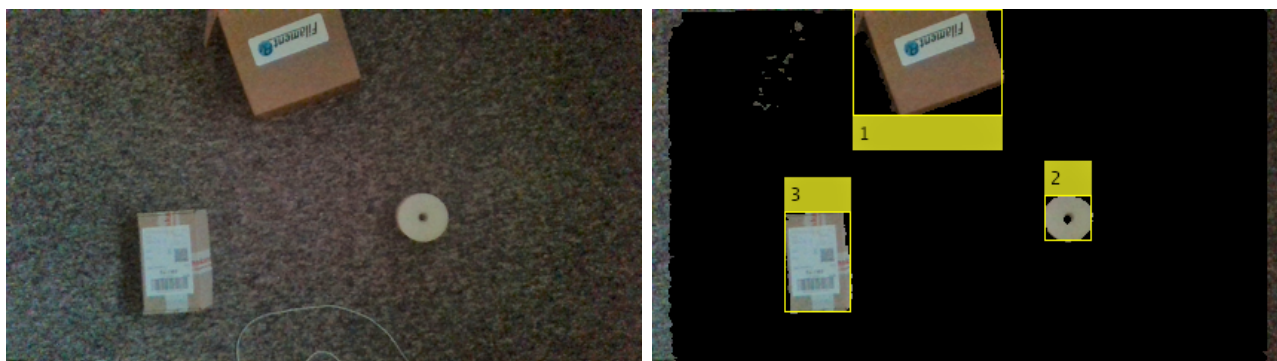


Obr. 4.3: Vyhľadávanie tvarov v PC. Vľavo pôvodný PC, vpravo PC zostavený z bodov, ktoré boli označené ako súčasť nájdeného tvaru

Poslednou metódou určenou pre spracovanie PC je rozdelenie objektov v PC podľa vzdialenosti bodov. Pre segmentáciu objektov je určená metóda *pc\_segment\_objects()*. Okrem PC, ktorý má

## 4 SPRACOVANIE DÁT ZO ZARIADENIA REALSENSE

byť rozdelený na jednotlivé objekty, sú metóde zadané ďalšie dva argumenty. Prvým je hodnota minimálnej vzdialenosti bodov, pri ktorej sú tieto body považované za body patriace rozdielnym telesám. Druhým argumentom je minimálny počet bodov, ktoré musí každý detekovaný objekt obsahovať, aby bol považovaný za validný. Pri používaní tejto metódy je odporúčané najskôr vyhľadať a odstrániť plochy ako sú steny alebo podlaha, ktoré by mohli objekty v priestore navzájom prepájať. Na obrázku 4.4 je možné vidieť samostatné nájdené objekty z hĺbkovej mapy. Pred samotným segmentovaním boli metódou *fit\_shape()* odstránené body odpovedajúce podlahe. Pre lepšie znázornenie boli pôvodný a filtrovaný PC prevedené na formát obrázku s označenými segmentovanými objektmi.



Obr. 4.4: Segmentácia PC na samostatné objekty. Vľavo zosnímaný obraz, vpravo obraz vytvorený po segmentovaní objektov podľa vzdialenosti

Zvyšné metódy RSIO slúžia najmä na ovládanie prehrávačov a prácu s dátami vo formáte obrázkov. Sú to pomocné metódy určené na zjednodušenie práce s dátami.

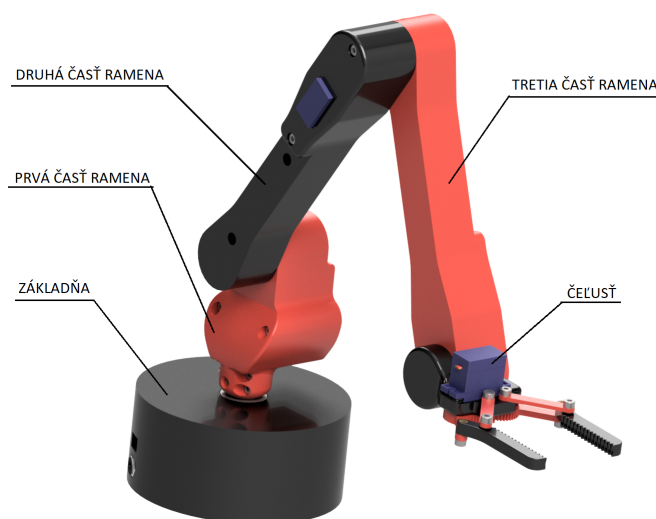
## 5 Návrh mechatronického exponátu

Napriek tomu, že návrh mechatronického exponátu nie je súčasťou zadania tejto práce, bolo potrebné navrhnuť a vyrobiť zariadenie, na ktorom je rozoznávanie objektov v hĺbkovej mape dobre využiteľné. Pri voľbe typu zariadenia bolo rozhodované medzi prenasledovacím dronom, zariadením, ktoré by sledovalo objekt daného tvaru a prípadne po tomto objekte strieľalo, a robotickým ramenom, ktoré by prenášalo objekty z jedného miesta na druhé, pričom by tieto objekty zároveň triedilo podľa tvaru a rozmerov. Keďže rýchlosť snímania a získavania PC je relatívne rýchla (približne 15 S/s), sú oba tieto návrhy zrealizovateľné. Avšak vplyvom algoritmov pre spracovanie PC za účelom hľadania objektov je rýchlosť snímania výrazne spomalená, čo by znemožňovalo rýchle sledovanie objektu a jeho prípadné zostrelenie. Pri výbere exponátu bolo teda nutné sústrediť sa na zariadenia, ktorých činnosť nie je znemožnená pomalým spracovávaním dát. S rastúcou vzdialenosťou taktiež rastie nepresnosť určovania polohy bodu v priestore. Z tohto dôvodu by musela byť kamera upevnená priamo na sledovacom drone. Keďže je kamera pripojená k počítaču USB káblom, musel by byť použitý buď extrémne dlhý USB kábel alebo by dron musel niesť celý počítač, na ktorom by prebiehali výpočty.

Po zvážení týchto podmienok bolo rozhodnuté, že ako exponát bude použité práve robotické rameno, ktoré bude presúvať a triediť objekty podľa ich tvaru a rozmeru.

### 5.1 Manipulátor verzia 1.0

Prvou verziou exponátu bolo robotické rameno - manipulátor so štyrmi stupňami voľnosti. Štyri stupne voľnosti značia, že má zariadenie štyri kĺby, ktoré je potrebné riadiť. Návrh manipulátoru je možné vidieť na obrázku 5.1. Model je vytvorený v študentskej verzii programu Autodesk Inventor 2018.



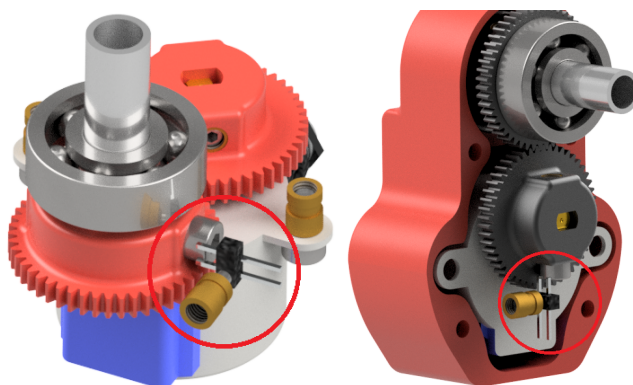
Obr. 5.1: Model prvej verzie manipulátoru

## 5 NÁVRH MECHATRONICKÉHO EXPONÁTU

Prvý kĺb slúži na rotáciu celého ramena okolo osi Z. Zvyšné kĺby tvoria spoje jednotlivých častí ramena a slúžia k jeho polohovaniu v rovine DZ. Os D leží v rovine XY a spája počiatok súradného systému manipulatoru s priemetom ťažiska presúvaného objektu do roviny XY.

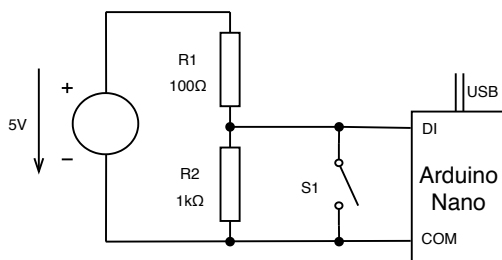
Pre ovládanie pohybu jednotlivých kĺbov boli využité servomotory a krokové motory. Rotácia okolo osi Z je v tejto verzii riešená použitím krokového motoru s označením 28BYJ-48-5V napájaného jednosmerným napätím o veľkosti 5 V. Rovnaký krokový motor bol použitý aj pre pohyb druhého kĺbu, keďže ten nesie najväčší záťažný moment. Zvyšné kĺby sú ovládané servomotormi Tower Pro™ SG90. Modely týchto motorov neboli vytvorené, ale stiahnuté z platformy *grabcad.com*. [29][30]

Riadiacou jednotkou zariadenia je mikrokontrolér Arduino Nano. Poloha servomotorov je určená pomocou riadiaceho signálu z mikrokontroléru. Určenie polohy krokových motorov je však obtiažnejšie. Pri zapnutí zariadenia je potrebné dostať krokové motory do základných pozícií, a následne počítať kroky a tie prevádzať na uhly. Určenie základných pozícií je riešené formou spínača, ako je možné vidieť na obrázku 5.2.



Obr. 5.2: Určovanie počiatočnej pozície využitím formy spínača

Spínač je zložený z dvoch pinov, ktoré tvoria rozpojený obvod. Pokiaľ sa servomotor natočí do polohy, kedy sa hlavička skrutky dotkne trčiacich pinov, je obvod spínača zopnutý a informácia o dosiahnutí referenčnej polohy je uložená. Od tohto okamihu sú Arduino počítané kroky motora. Na obrázku 5.3 je tento spínač označený ako S1.



Obr. 5.3: Schéma obvodu využívaného pri určovaní základnej polohy krokového motora

Pokiaľ motor nie je v referenčnej polohe, je spínač rozpojený a prúd tečie cez oba odpory. Potenciál v snímanom uzle určuje pomer odporov R1 a R2, čiže vstupné napätie je rozdelené na napätia odporov v pomere 1:10. Potenciál v snímanom uzle je v tomto prípade približne 4.5 V, čo odpovedá logickej úrovni 1 na digitálnom vstupe Arduina. V okamihu kedy motor dosiahne referenčnú polohu, je spínač zopnutý, čím je vyskratovaný odpor R2 a potenciál v snímanom bode je nulový, čo odpovedá logickej úrovni 0 na digitálnom vstupe Arduina.

Na voľnom konci manipulátoru sa nachádza čeľusť slúžiaca k uchopovaniu objektov. Detail čeľuste je na obrázku 5.4. Čeľusť je v tejto verzii riešená formou dvoch štvorkĺbových mechanizmov ovládaných servomotorom. Každý z mechanizmov pohybuje jednou z dvojice čeľustí, pričom jeden z týchto mechanizmov je riadený priamo servomotorom a druhý pomocou ozubeného prevodu, čo zaisťuje symetrický pohyb čeľustí. Dĺžka ramien mechanizmu a tvar čeľuste zaisťujú premenu rotačného pohybu na translačný, a tak stálu rovnobežnosť uchopovacích plôch.



Obr. 5.4: Mechanizmus čeľustí prvej verzie manipulátoru

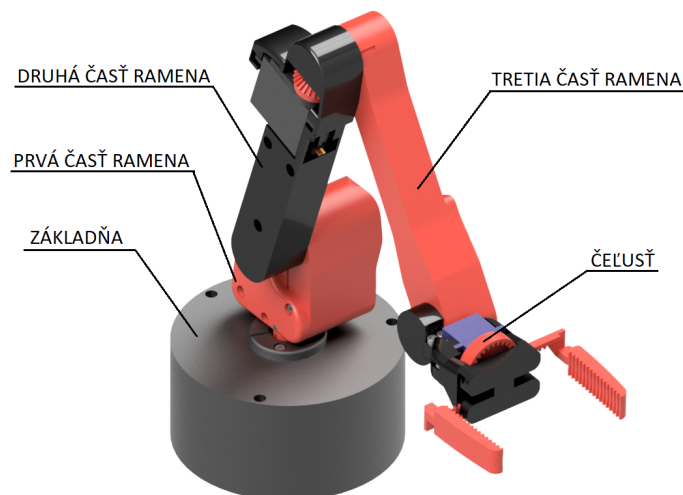
Táto verzia manipulátoru má však, ako sa ukázalo po jeho vyrobení a zostavení, dva zásadné problémy. Prvým nie až tak závažným problémom je slabá tuhosť mechanizmu čeľustí, čo spôsobuje občasné vypadávanie uchopených objektov. Závažnejším problémom však je fakt, že pri pohybe kĺbov sú motory v druhom a treťom kĺbe zatažené maximálnym momentom, ktorý tieto motory dokážu vyvolať, čím sa stáva pohyb manipulátoru nestabilný a manipulátor sa bortí pod vlastnou váhou.

Za účelom odstránenia týchto nedostatkov je navrhnutá nová verzia manipulátoru.

### 5.2 Manipulátor verzia 2.0

Na základe chýb odhalených v návrhu prvej verzie bola navrhnutá nová verzia manipulátoru. V tejto verzii sú použité silnejšie servomotory pre riadenie pohybu jednotlivých kĺbov a je vytvorený nový mechanizmus čeľustí. Nové servomotory sú oveľa väčšie ako motory použité v predchádzajúcej verzii, kvôli čomu nie je možné použiť návrh z prvej verzie a manipulátor musel byť od základu prerobený. Nová verzia, rovnako ako stará, je taktiež vytvorená v študentskej verzii programu Autodesk Inventor 2018. Oba krokové motory sú v tejto verzii nahradené servomotorami MG995, ktoré pri napájaní napätím 5 V dosahujú krútiaci moment až 1.5 Nm. Takisto aj servomotor v treťom kĺbe predchádzajúcej verzie je nahradený silnejším servomotorom modelu MG946R s približne desaťnásobným krútiacim momentom ako pri použití servomotoru SG90.

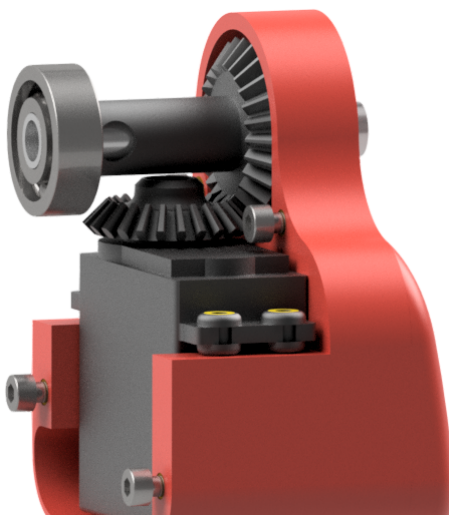
Na základe väčších rozmerov motora ukrytého v základni manipulátoru je táto časť výrazne zväčšená, čo umožňuje vloženie riadiacej jednotky priamo do konštrukcie.



Obr. 5.5: Model druhej verzie manipulátoru

Servomotory použité v tomto návrhu sú schopné natočenia v rozmedzí  $0^\circ$  až  $180^\circ$ . Pre dosiahnutie rotácie okolo osi Z v rozsahu  $0^\circ$  až  $360^\circ$  je výstupná hriadeľ servomotora sprevodovaná čelným ozubeným súkolesím s prevodovým pomerom  $i = 2$ . Tento prevod síce zmenší krútiaci moment motora na polovicu, ale umožňuje zariadeniu urobiť plnú otáčku okolo osi Z. Prevody ozubenými súkolesiami sú použité aj v predchádzajúcej verzii, kde sú prevodové pomery všetkých súkolesí  $i = 1$ . Použitie prevodu namiesto priameho pripojenia ďalšej časti zariadenia k servomotoru umožňuje vedenie kabeláže osou otáčania jednotlivých kĺbov. Vďaka tomu je možné viesť všetku kabeláž k motorom vnútorom jednotlivých častí manipulátoru. Súčasťou základne sú dve LED diódy indikujúce stav zariadenia.

Výrazne sú pozmenené aj prvé dve časti ramena vystupujúce zo základne, kde výstupná os servomotora a os rotácie kĺbu zvierajú  $90^\circ$  uhol. Pre prenos krútiaceho momentu je použitý prevod kužeľovým ozubeným súkolesím, ako je možné vidieť na obrázku 5.6.



Obr. 5.6: Kužeľové ozubené súkolesie použité pre prenos krútiaceho momentu

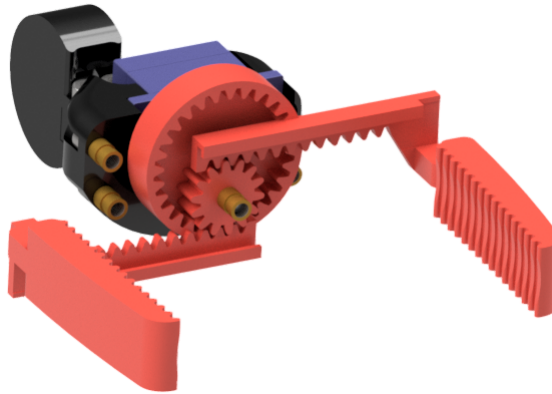
Duté valcové profily prenášajúce moment medzi základňou a prvou časťou ramena a medzi prvou a druhou časťou ramena sú uložené v radiálnych guľôčkových ložiskách, ktoré nesú zaťaženie ďalších častí ramena.



## 5 NÁVRH MECHATRONICKÉHO EXPONÁTU

Servomotor v druhej časti ramena (medzi druhým a tretím kĺbom) je čiastočne vysunutý mimo šasi časti ramena. Vďaka tomu je motor dostatočne chladený. Motor tvorí spoj medzi druhým a tretím kĺbom, kde je zo strany druhého kĺbu pripojený k dutému telu časti ramena, v ktorom sú ukryté káble, a z druhej strany je výstupná hriadeľ pripojená ku kuželovému súkolesiu, ktoré umožňuje prenesenie krútiaceho momentu do osi rotácie tretieho kĺbu.

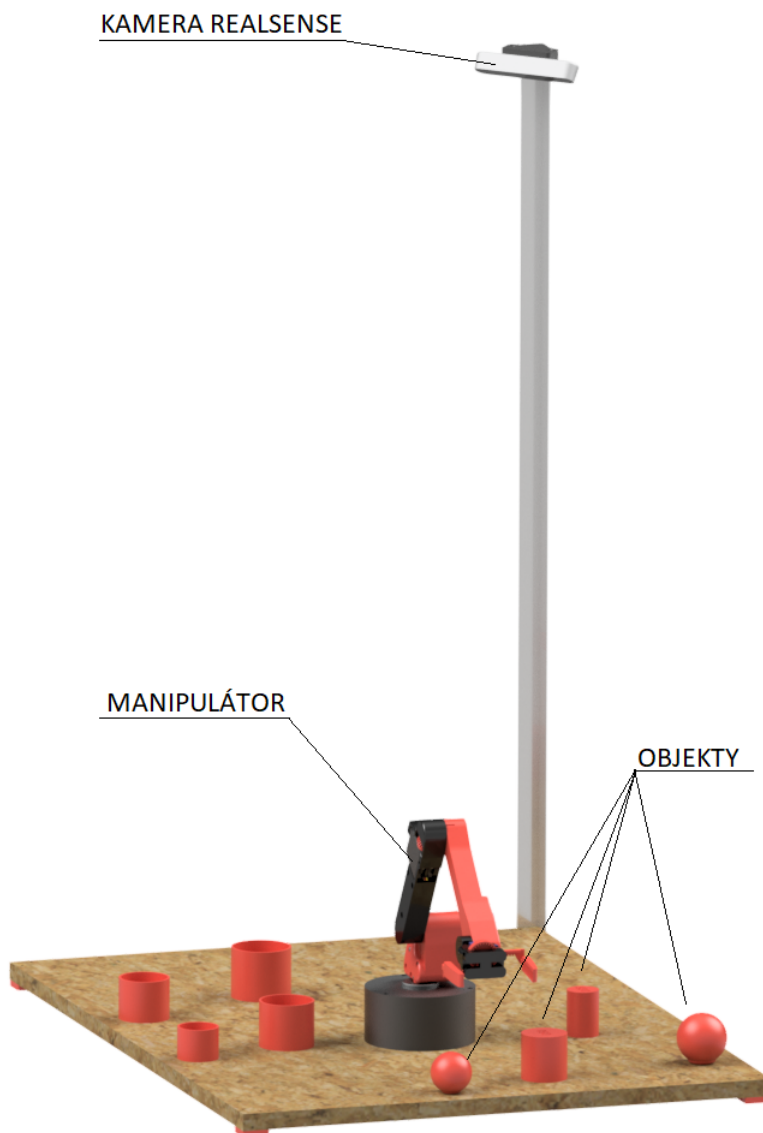
Poslednou zmenou v návrhu je čelusť. V novej verzii manipulátoru pozostáva mechanizmus čeluste z komplikovanejšieho prevodu ozubenými kolesami. Krútiaci moment servomotora je prenesený na pastorok prevodu použitím súkolesia s vnútorným ozubením. Použitím vnútorného ozubenja je umožnené zmenšenie rozmerov čelustí. Rotačný pohyb pastorku je následne premenený na translačný pohyb použitím ozubených hrebeňov, ako je možné vidieť na obrázku 5.7.



Obr. 5.7: Mechanizmus prevodu rotačného pohybu na translačný v čelustiach modelu 2.0

Pri použití prevodu ozubeným hrebeňom je lineárny posuv na zub určený vzťahom  $l = m \cdot \pi [mm]$ , kde  $m$  je modul ozubenja. V tomto prípade je modul ozubenja  $m = 1$ , čo znamená, že pri pootočení pastorka o zub je čelusť vysunutá približne o 3.14 mm. Keďže sú použité dva symetricky sa pohybujúce hrebene, sú čeluste pri pootočení pastorka o jeden zub roztvorené o približne 6.28 mm. Za účelom dosiahnutia maximálneho rozpätia čelustí aspoň 70 mm je použitý pastorok s počtom zubov  $z_1 = 14$ . Keďže sa servomotor môže otočiť maximálne o  $180^\circ$ , je zvolený prevodový pomer čelného ozubenja  $i = 2$ , čiže hnacie ozubené koleso s vnútornými zubmi má počet zubov  $z_2 = 28$  a celkový priemer hnacieho kolesa je  $D_2 = 33.55$  mm. Touto konštrukciou je dosiahnuté výsledné rozpätie čelustí v rozmedzí 2.5 až 79 mm.

Manipulátor je následne upevnený na dosku, ku ktorej je v jednom rohu pripevnený hliníkový L profil, s dĺžkou 1 m, na ktorého druhom konci je umiestnená kamera RealSense snímajúca celú scénu (obrázok 5.8). Prenášanými telieskami sú dva valce s priermi  $D_1 = 35$  mm a  $D_2 = 50$  mm a dve gule o prieroch  $SD_1 = 45$  mm a  $SD_2 = 55$  mm. Úlohou manipulátoru je preniesť tieto objekty do zásobníkov, dutých valcov s vnútornými priermi zväčšenými pre každé teleso o 5 mm a výškou rovnou priemeru telesa, pre ktoré je zásobník určený.



Obr. 5.8: Manipulátor postavený na podložke s držiakom kamery, telieskami a ich zásobníkmi

### 5.3 Výroba manipulátoru

Jadrom kĺbov manipulátoru sú duté kruhové hliníkové tyče, ktoré prenášajú krútiaci moment medzi jednotlivými časťami manipulátoru.

Zvyšné časti modelu sú vyrobené technológiou 3D tlače. Príprava tlače navrhnutých objektov prebiehala v programe Simplify3D, v ktorom boli telesá „nakrájané“ a následne bol vygenerovaný súbor obsahujúci riadiaci kód pre tlačiareň. Výhodou 3D tlače sú nízke náklady na samotnú výrobu a možnosť výroby objektov takmer ľubovoľného tvaru. Pre tlač bol použitý materiál PLA. Nesmiernym uľahčením práce je možnosť vytvoriť sieťový povrch bez toho, aby ho bolo potrebné modelovať. Stačí zvoliť tlač bez plných povrchových vrstiev telesa, zvoliť vhodný tvar a hustotu výplne a sieť, ktorá bude tvoriť povrch objektu po vytlačení, je automaticky vygenerovaná programom. Sieťový povrch objektov má dve výhody. Prvou je odľahčenie telesa, ktorého hmotnosť je už aj tak dosť nízka, keďže je vyrobené z plastu. Druhou výhodou



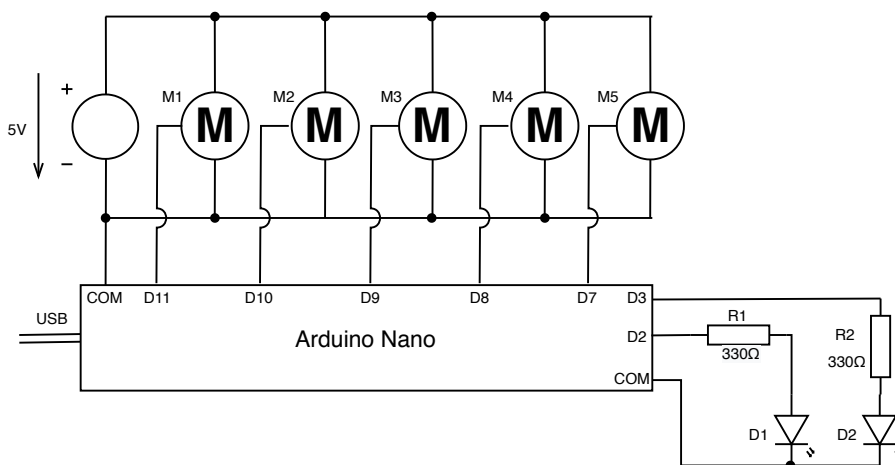
sieťového povrchu objektov je možnosť chladenia motorov umiestnených v týchto objektoch. Sieťový povrch zabezpečuje dostatočný prietok vzduchu odvádzajúceho teplo z okolia motora.

Väčšinu spojov, ktoré spájajú jednotlivé časti manipulátoru, tvoria skrutkové spoje. Do predpripravených dier v telesách boli po vytlačení za tepla vlisované závitové vložky zastávajúce funkciu matice. Po vychladnutí je tak vytvorený plnohodnotný lisovaný spoj plastovej súčiastky a mosadznej závitovej vložky.

### 5.4 Elektrické zapojenie

Mozgom manipulátoru je mikrokontrolér Arduino Nano. Jadrom mikrokontroléru je 8-bitový procesor ATmega328 naktovaný na frekvenciu 16 MHz. Mikrokontrolér má 22 I/O pinov, z čoho 8 pinov slúži ako analogový vstup a 6 pinov je možné použiť ako analógový výstup (PWM). Arduino Nano môže byť napájaný jednosmerným napätím v rozsahu 5 až 12 V. Mikrokontrolér manipulátoru je napájaný napätím 5 V z USB pripojeného k počítaču. [31]

K mikrokontroléru sú pripojené dve LED diódy indikujúce stav manipulátoru. Zelená dióda značí, že je manipulátor pripojený a pripravený na použitie a červená značí, že je manipulátor zanepraznadený činnosťou. Schéma zapojenia je na obrázku 5.9

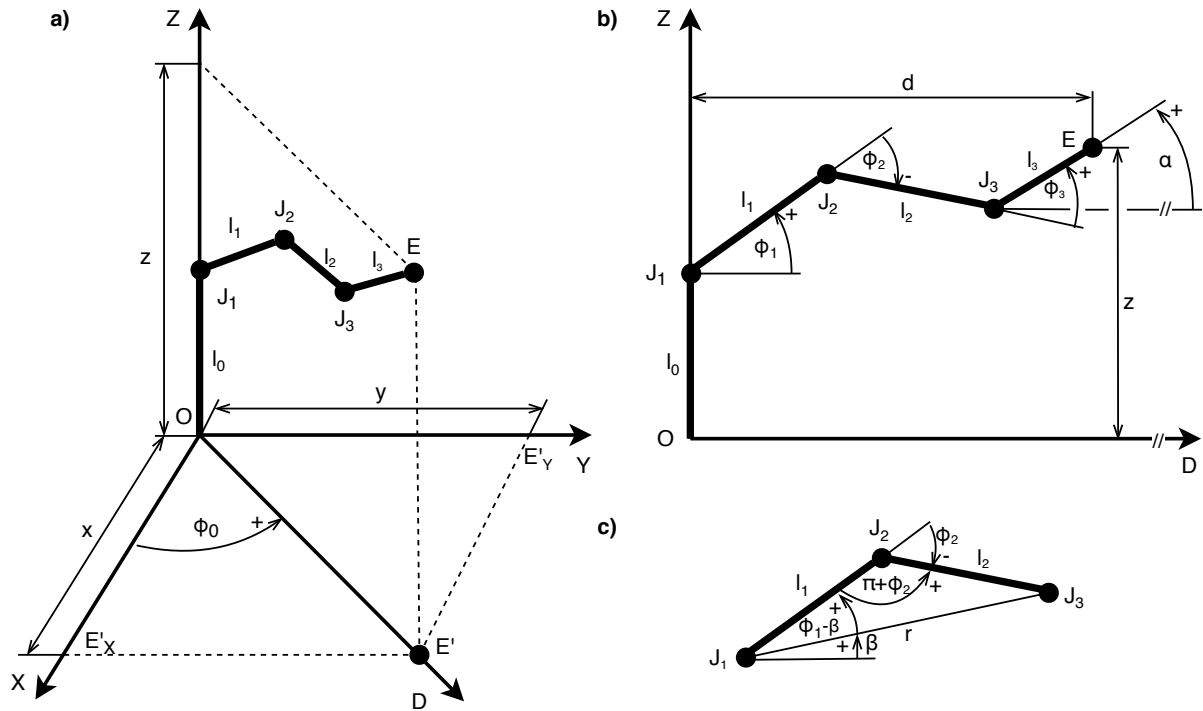


Obr. 5.9: Schéma elektrického zapojenia manipulátoru

Ako aktuátory sú použité servomotory MG995, MG946R a Tower Pro™ SG90. Servomotory sa skladajú z jednosmerného DC motoru, vstavanej prevodovky znižujúcej otáčky a zvyšujúcej krútiaci moment motora a vnútorného riadiaceho čipu. Výstupná hriadeľ je pripojená k potenciometru, ktorý indikuje polohu servomotoru. Keďže väčšina rotačných potenciometrov má uhol natočenia mierne viac ako 200°, je maximálne natočenie samotného servomotoru obmedzené na 180°. Pre prácu so servomotormi je potrebné napájať ich z externého zdroja, keďže každý servomotor dosahuje pri maximálnom krútiacom momente výkon približne 5 W a maximálny výstupný výkon Arduina je 100 mW. Servomotory sú riadené PWM signálom o frekvencii 50 Hz. Poloha je riadená striedou PWM signálu s pracovným cyklom v rozsahu približne 2 až 15 %. Limity pracovného cyklu sa pri rôznych modeloch líšia a je ich potrebné nastaviť pri kalibrácii servomotoru. [32]

Manipulátor je napájaný zdrojom jednosmerného napätia s výstupom 5 V, 25 W.

## 5.5 Výpočet uhlov natočenia klbov a servomotorov



Obr. 5.10: Náčrt manipulátoru s označenými dĺžkami ramien a uhlami; a) v karteziánskom súradnicovom systéme, b) priemet do roviny DZ, c) náčrt pre výpočet pomocou kosínovej vety

Na obrázku 5.10 je zjednodušený náčrt kostry manipulátoru s označenými dĺžkami strán a uhlami. Body  $J_1$ ,  $J_2$  a  $J_3$  sú klby medzi jednotlivými časťami ramena, bodom  $E$  sú označené čeluste. Uhol  $\phi_0$  je uhol natočenia prvého servomotora umožňujúceho otáčanie ramena okolo osi  $Z$ . Uhol  $\alpha$  je uhol, pod ktorým je objekt čelustami uchopený. Aby bola sústava riešiteľná, je potrebné ho určiť spolu so súradnicami ( $X$ ,  $Y$  a  $Z$ ) objektu. Určenie uhlu  $\alpha$  je zaistené softwarom, ktorý vypočítava uhly natočenia klbov manipulátoru. Prednostne je overené, či je možné dostať sa k objektu pod uhlom  $\alpha$   $0^\circ$  alebo  $-90^\circ$ . Pokiaľ to možné nie je, je postupne overované dosiahnutie objektu pod uhlami v rozmedzí  $-90^\circ$  až  $0^\circ$ . Dĺžky ramien sú pevne stanovené modelom.

Pre výpočet uhlov  $\phi_0$  až  $\phi_4$  je potrebné určiť sústavu rovníc a poznať súradnice bodu  $X$ ,  $Y$ ,  $Z$  a uhol  $\alpha$ .

$$d = \sqrt{x^2 + y^2} \quad (5.1)$$

Vzťah 5.1 vyplýva z Pytagorovej vety, vzdialenosť  $d$  je dĺžka prepony trojuholníka  $OE'E'_X$ , kde bod  $E'$  je priemetom bodu  $E$  na rovine  $XY$ , bod  $E'_X$  je priemet bodu  $E'$  na osi  $X$  a bod  $O$  je počiatok súradnicového systému. Vyplývajúci z trojuholníka  $OE'E'_X$  je možné určiť uhol  $\phi_0$  vzťahom 5.2.

$$\phi_0 = \arctan\left(\frac{y}{x}\right) \quad (5.2)$$

## 5 NÁVRH MECHATRONICKÉHO EXPONÁTU

Rozmery

$$d = l_1 \cdot \cos(\phi_1) + l_2 \cdot \cos(\phi_1 + \phi_2) + l_3 \cdot \cos(\phi_1 + \phi_2 + \phi_3) \quad (5.3)$$

$$z = l_0 + l_1 \cdot \sin(\phi_1) + l_2 \cdot \sin(\phi_1 + \phi_2) + l_3 \cdot \sin(\phi_1 + \phi_2 + \phi_3) \quad (5.4)$$

$$\alpha = \phi_1 + \phi_2 + \phi_3 \quad (5.5)$$

Existuje však jednoduchší spôsob výpočtu uhlov  $\phi_1$  a  $\phi_2$  než vyjadrovaním a dosádzaním do rovníc 5.3 a 5.4. Po prekreslení ramien  $l_1$  a  $l_2$  z obrázku 5.10b, vytvorení trojuholníka  $J_1J_2J_3$  a odvodení chýbajúcich uhlov pri vrcholoch  $J_1$  a  $J_2$  (5.10c) je možné k výpočtu uhlov  $\phi_1$  a  $\phi_2$  použiť kosínovú vetu (vzťahy 5.7 a 5.8). Dĺžka ramena  $r$  je určená vzťahom 5.6.

$$r = \sqrt{(d - l_3 \cdot \cos \alpha)^2 + (z - l_0 - l_3 \cdot \sin \alpha)^2} \quad (5.6)$$

$$r = \sqrt{l_1^2 + l_2^2 - 2 \cdot l_1 \cdot l_2 \cdot \cos(\pi + \phi_2)} \quad (5.7)$$

$$l_2 = \sqrt{l_1^2 + r^2 - 2 \cdot l_1 \cdot r \cdot \cos(\phi_1 - \beta)} \quad (5.8)$$

Uhol  $\beta$  je definovaný vzťahom 5.9.

$$\beta = \arctan\left(\frac{z - l_0 - l_3 \cdot \sin \alpha}{d - l_3 \cdot \cos \alpha}\right) \quad (5.9)$$

Úpravou vzťahu 5.8 a dosadením vzťahu 5.9 namiesto  $\beta$  je získaný vzťah pre výpočet uhlu  $\phi_1$  5.10, kde  $r$  je dané vzťahom 5.6.

$$\phi_1 = \arctan\left(\frac{z - l_0 - l_3 \cdot \sin \alpha}{d - l_3 \cdot \cos \alpha}\right) + \arccos\left(\frac{r^2 + l_1^2 - l_2^2}{2 \cdot r \cdot l_1}\right) \quad (5.10)$$

Uhol  $\phi_2$  je vyjadrený zo vzťahu 5.7 a definovaný vzťahom 5.11.

$$\phi_2 = \arccos\left(\frac{l_1^2 + l_2^2 - (d - l_3 \cdot \cos \alpha)^2 - (z - l_0 - l_3 \cdot \sin \alpha)^2}{2 \cdot l_1 \cdot l_2}\right) - \pi \quad (5.11)$$

## 5 NÁVRH MECHATRONICKÉHO EXPONÁTU

Vzťah pre určenie uhlu  $\phi_3$  je získaný úpravou a vyjadrením uhlu  $\phi_3$  z 5.5.

$$\phi_3 = \alpha - \phi_1 - \phi_2 \quad (5.12)$$

Uhol servomotoru ovládajúceho čeluste je závislý na polomeri  $R$  objektu a je určený vzťahom 5.13, kde  $D_{min}$  je minimálne rozpätie čelustí  $D_{min} = 0.0225$  m,  $i$  je prevodový pomer čelného ozubeného súkolesia  $i = 2$  a  $z_1$  je počet zubov na pastorku  $z_1 = 14$ .

$$\phi_4 = \frac{2 * R - D_{min}}{i \cdot z_1} \quad (5.13)$$

Po dosadení konštánt do vzťahu 5.13 vznikne vzťah závislý na polomere objektu  $R$ .

$$\phi_4 = \frac{1}{14} \cdot R - \frac{45}{56} \quad (5.14)$$

### 5.6 Software a simulácia

Komunikácia s manipulátorom je sprostredkovaná prostredníctvom objektu *Manipulator* (MO) vytvoreného v prostredí MATLAB. Pri inicializácii MO je vytvorený objekt *arduino*, ktorý zabezpečuje komunikáciu s mikrokontrolérom Arduino Nano. Okrem komunikačnej linky sú pri inicializácii načítané dáta zo súborov s koncovkou *.stl*. V týchto súboroch sú uložené jednotlivé časti modelu manipulátoru vytvoreného v programe Autodesk Inventor. Po načítaní týchto súborov v MO je vytvorený simulačný model manipulátoru. Ďalej sú extrahované základné rozmery jednotlivých komponent, z ktorých sú následne vytvorené kolízne objekty formátu *collisionBox*, *collisionSphere* a *collisionCylinder*. Narozdiel od objektov vytvorených po načítaní z *.stl* súborov sú kolízne objekty zjednodušené na jeden zo základných tvarov hranol, valec alebo guľa. Pre prácu s týmito objektmi slúži knižnica *Robotics System Toolbox*, ktorej funkcie umožňujú kontrolu vzájomných kolízií jednotlivých kolíznych objektov, a tak aj zistiť prípadnú kolíziu manipulátoru.

V MO prebieha taktiež výpočet uhlov jednotlivých ramien použitím vzťahov 5.2, 5.10, 5.11, 5.12 a 5.14 popísaných v kapitole 5.5. Tieto uhly však nezodpovedajú priamo polohám servomotorov, preto ich je potrebné ďalej upraviť. Prevod vypočítaných uhlov na polohy servomotorov je popísaný vzťahmi 5.15 až 5.19.

$$\theta_0 = -\frac{\phi_0}{2} \quad (5.15)$$

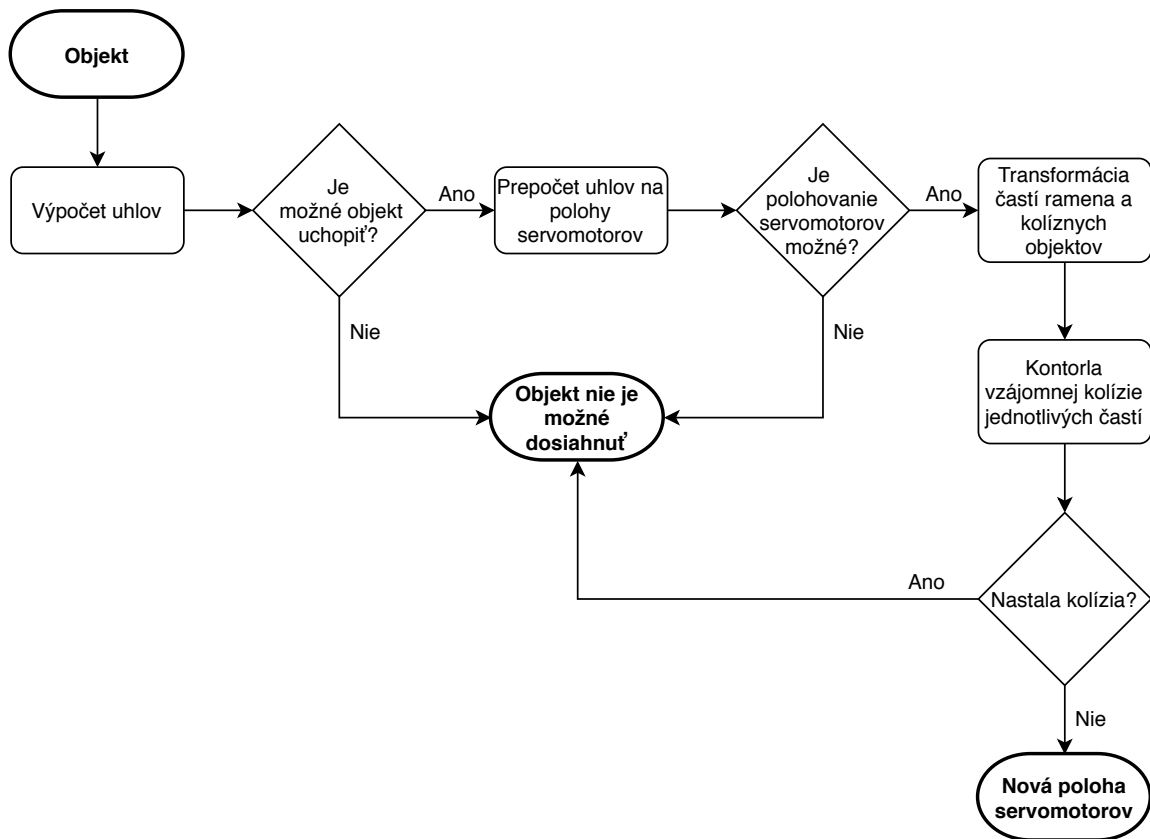
$$\theta_1 = \phi_1 + \frac{\pi}{2} \quad (5.16)$$

$$\theta_2 = -\phi_2 \quad (5.17)$$

$$\theta_3 = \frac{\pi}{2} - \phi_3 \quad (5.18)$$

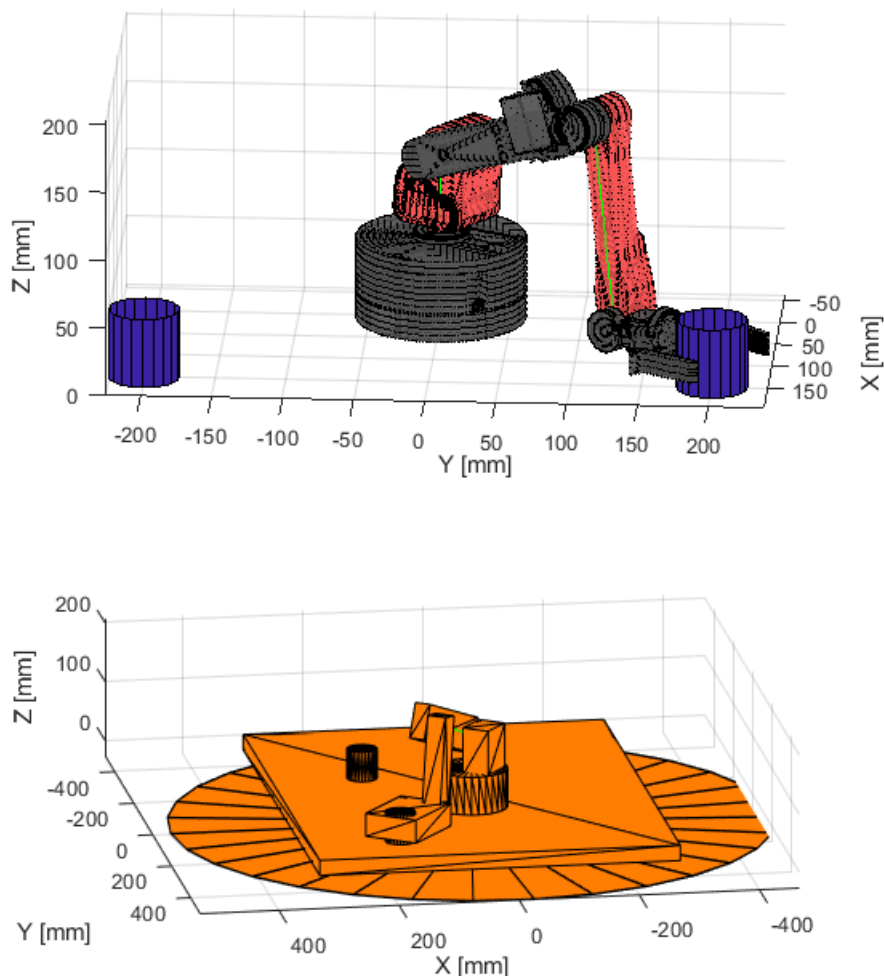
$$\theta_4 = \pi - \phi_4 \quad (5.19)$$

Po inicializovaní MO je manipulátor pripravený k činnosti. Novú polohu je možné určiť dvoma spôsobmi. Prvým spôsobom je ručné zadanie polôh jednotlivých servomotorov, čo je využívané najmä k ručnému polohovaniu. Druhým spôsobom je zadanie objektu, ku ktorému sa má manipulátor dostať. Pre dosiahnutie objektu je potrebné definovať polohu jeho stredu v priestore a jeho polomer. Vhodným formátom objektu je formát *cylinderModel* alebo *sphereModel*. Objekt definovaný v tomto formáte je výstupom metódy *fit\_shape()* z RSIO. Pribeh simulácie novej polohy manipulátoru je popísaný na diagrame 5.11



Obr. 5.11: Diagram popisujúci priebeh kontroly pred pohybom manipulátoru

Pre výpočet uhlov je potrebné zadať uhol  $\alpha$ , pod ktorým má byť objekt uchopený. Určenie tohto uhlu je automaticky prevedené metódou MO *check\_object()*, ktorý určí vhodný uhol uchopenia objektu v závislosti na aktuálnej polohe nasnímaného objektu a jeho novej polohe (polohe kam má byť objekt presunutý). Určením vhodného uhlu  $\alpha$  pre aktuálnu a novú polohu je získaný uhol, pod ktorým je následne objekt uchopený a následne položený na novú pozíciu. Možnosť uchopiť objekt pod uhlom  $\alpha$  rovným 0 alebo  $-\frac{\pi}{2}$  je kontrolovaná prednostne, pokiaľ nie je možné objekt uchopiť pod jedným z týchto uhlov je následne otestovaná možnosť uchopenia objektu pod uhlom  $\alpha \in \langle -\frac{\pi}{2}, 0 \rangle$ .



Obr. 5.12: Simulácia polohy manipulátoru pre dosiahnutie objektu

Na obrázku 5.12 je možné vidieť simuláciu vypočítanej polohy manipulátoru za účelom dosiahnutia objektu. Objekt je v tomto prípade valec s polomerom 25 mm a výškou 50 mm. Objekt, na ktorý je manipulátor sústredený, je objekt, ktorý má byť presunutý a druhý objekt v simulácii značí polohu, kam má byť objekt položený. Na spodnom obrázku je simulácia v podaní kolíznych objektov. V tomto prípade kolízia nenastala a objekt je dosiahnuteľný.

Výstupom simulácie je logická hodnota reprezentujúca možnosť daný objekt dosiahnuť. Pokiaľ je daný objekt v dosahu manipulátoru, je možné použiť metódy MO určené pre pohyb manipulátoru. Jedná sa o metódy *move\_base()* a *move\_arm()*, ktoré okrem samotného pohybu manipulátoru umožňujú aj reguláciu rýchlosti pohybu. Rýchlosť pohybu je zadávaná argumentom metódy v jednotkách  $[deg \cdot s^{-1}]$ .

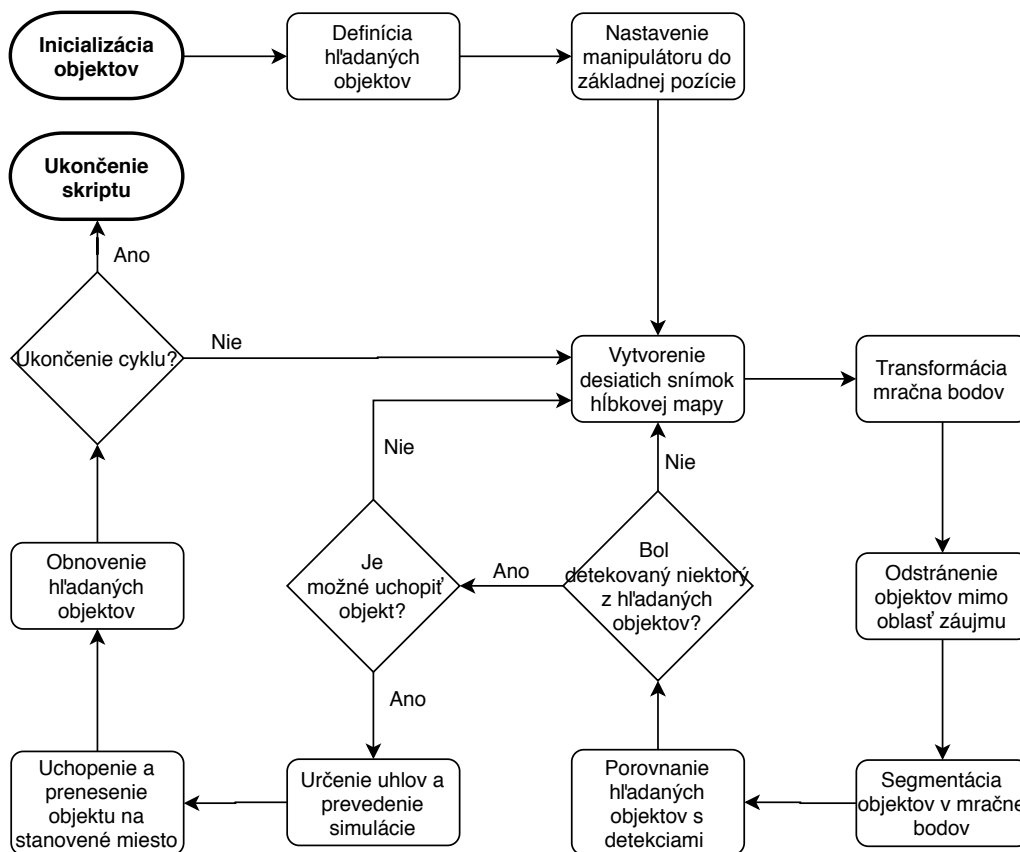
Poloha servomotoru je softwarovo zmenená okamžite po zadaní príkazu na zmenu, aj keď servomotor v skutočnosti danú polohu ešte nedosiahol. Tento jav je zapríčinený spôsobom, ktorým je aktuálna poloha servomotoru získavaná. Aj napriek tomu, že servomotory obsahujú potenciometer, ktorý indikuje uhol natočenia výstupnej hriadele, nie je pomocou neho získavaná polohová odozva. Polohovú odozvu určuje strieda nastavená na signálovom pine mikrokontroléru. Vďaka tomu je možné spustiť pohyb všetkých servomotorov súčasne.

Keďže je rýchlosť otáčania servomotorov veľmi veľká je ju potrebné regulovať. Regulácia je

možná dvomi spôsobmi. Prvým zo spôsobov je regulácia pomocou napájacieho napätia, čím sa však zároveň mení krútiaci moment. Druhým spôsobom je softwarová regulácia. V tomto prípade je využívaná práve druhá metóda.

Odčítaním aktuálnej pozície servomotorov od novej pozície je určený rozdiel polôh. Z týchto rozdielov je vybraný najväčší, ktorý je porovnaný s maximálnym povoleným krokom určeným parametrom MO. Najväčší rozdiel polôh je vydelený týmto parametrom, čím je získaný počet potrebných medzikrokov. Všetky rozdiely polôh sú vydelené počtom medzikrokov, čím je zaistený plynulý presuv manipulátoru na určené miesto. Najväčšia hodnota medzikroku je vydelená rýchlosťou určenou argumentom metód *move\_arm()* alebo *move\_base()*, čím je získaný čas pauzy medzi jednotlivými medzikrokmi. Časovaním jednotlivých medzikrokov je regulovaná rýchlosť pohybu manipulátoru.

## 5.7 Implementácia snímania scény a spracovania obrazu do exponátu

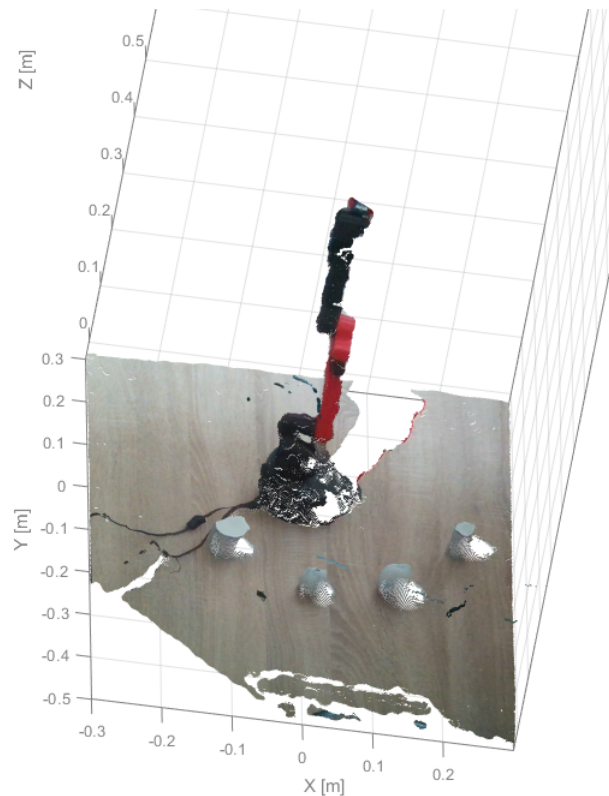


Obr. 5.13: Diagram priebehu riadiaceho programu exponátu

Objekty vytvorené v tejto práci sú spojené v programe predstavujúcim implementáciu jednotlivých častí do celku. Priebeh tohto programu je popísaný diagramom na obrázku 5.13. Pri inicializácii programu sú vytvorené objekty RSO, RSIO. Zapnutý je aj dátový tok z farebného a hĺbkového senzoru. Inicializované filtre sú použité pre postprocesovanie mračna bodov. Účelom exponátu je nájsť na scéne štyri objekty. Týmito objektmi sú dve gule a dva valce rôznych priemerov. Priemery týchto objektov sú zadané do programu, umožňujúc tak presnejšie rozpoznanie objektov nájdených v mračne bodov. Následne je inicializovaný MO a manipulátor je nastavený do východzej pozície tak, aby čo najmenej ovplyvňoval získané hĺbkové dáta. Hĺbkový senzor je

## 5 NÁVRH MECHATRONICKÉHO EXPONÁTU

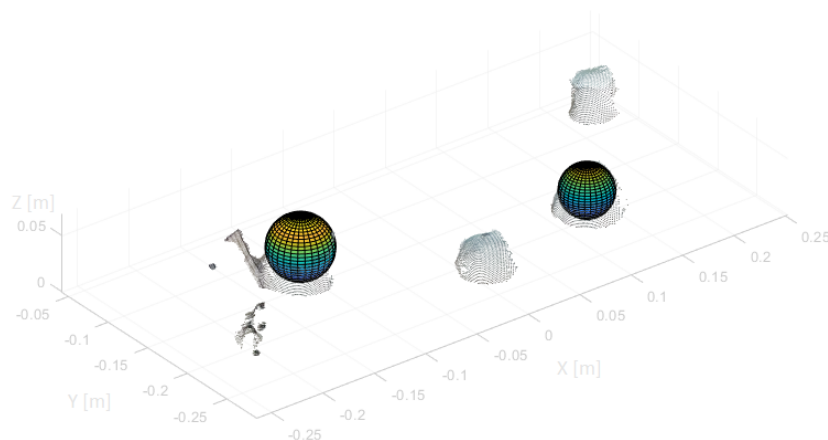
nastavený na presné snímanie (nastavenie *visual\_preset*) a hĺbkové jednotky sú nastavené na hodnotu 0.0001, čo redukuje maximálnu merateľnú vzdialenosť na približne 6.5 m s presnosťou získaných dát o polohe bodov na 0.1 mm.



Obr. 5.14: Zosnímanie transformovanie mračna bodov

Po ukončení inicializácie je vytvorených niekoľko snímok potrebných pre stabilizáciu filtrov a zlepšenie kvality výsledného mračna bodov, ako je možné vidieť na obrázku 5.14.

Po získaní dát zo senzoru RealSense sú hĺbkové a farebné dáta prevedené na formát *pointCloud* a ďalej spracované. Odstránením nepotrebných bodov je urýchlená segmentácia objektov z mračna. Z nájdených objektov sú odstránené tie, v ktorých počet bodov nedosiahol minimálnu hranicu, čím je odstránený šum.



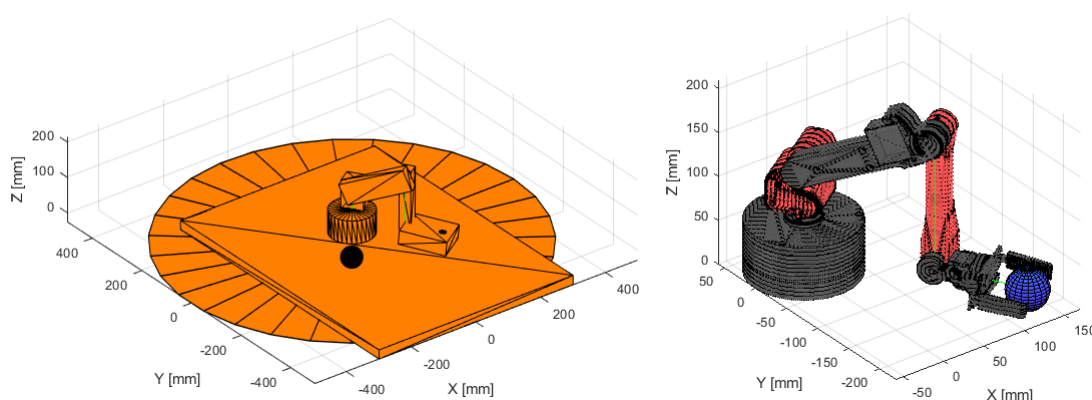
Obr. 5.15: Odstránenie bodov mimo oblasť záujmu, segmentácia a klasifikácia objektov v snímanej scéne

Pre vyhľadanie objektov požadovaných tvarov a rozmerov z detekcií je použitá metóda RSIO



*fit\_shape()*, ktorá má za úlohu v danom mračne bodov nájsť tieto objekty. Výstupom metódy sú objekty vo formáte dátového typu *cylinderModel* alebo *sphereModel*, ktoré obsahujú informácie o polohe a rozmeroch detekovaného objektu. Každý z detekovaných objektov je porovnávaný s hľadanými objektmi. Pokiaľ je detekovaný objekt zhodný s viacerými hľadanými objektmi, je porovnávaná priemerná odchýlka bodov od jednotlivých vytvorených modelov. Model s najmenšou odchýlkou bodov je považovaný za správnu detekciu. Výsledok detekcie je možné vidieť na obrázku 5.15.

Modely hľadaných objektov sú následne predané objektu MO, ktorý určí, či je možné manipulátorom daný objekt uchopiť. Pri určovaní či je alebo nie je možné daný objekt uchopiť, je posudzovaná jeho poloha (obrázok 5.16 vpravo), vzájomná kolízia jednotlivých častí manipulátoru a kolízia s ostatnými detekovanými objektmi (obrázok 5.16 vľavo). Pokiaľ je objekt v dosahu manipulátoru a nenastane žiadna kolízia, je objekt manipulátorom uchopený a premiestnený (obrázok 5.17).



Obr. 5.16: Kontrola kolízie (vľavo) a simulácia dosiahnutia objektu (vpravo)

V dôsledku malej pravdepodobnosti faktu, že by bolo objekt možné uchopiť a následne premiestniť na miesto určené iným objektom pri rovnakom uhle úchopu je funkcionálnosť manipulátoru pozmenená. Uchopený objekt je presunutý na súradnice určené nasledujúcimi podmienkami. Pokiaľ je detekovaný objekt guľa, je objekt presunutý na súradnice  $X_2 = -|X_1|$ ;  $Y_2 = -Y_1$ ;  $Z_2 = Z_1$ , pokiaľ je objektom valec, je objekt presunutý na súradnice  $X_2 = |X_1|$ ;  $Y_2 = -Y_1$ ;  $Z_2 = Z_1$ , kde  $X_1$ ,  $Y_1$  a  $Z_1$  sú súradnice stredu detekovaného objektu. Objekty sú po presunutí roztriedené, pričom deliacu rovinu medzi nimi tvorí rovina YZ.



Obr. 5.17: Manipulátor zdvíhajúci objekt po úspešnom ukončení simulácie

Po presunutí všetkých nájdených objektov zo snímky je manipulátor opäť nastavený do počiatočnej pozície, snímanie scény je znovu spustené a cyklus sa opakuje.

# Záver

Táto práca je zameraná na získavanie a spracovanie dát zo zariadení RealSense, spracovanie mračien bodov, triedenie a určovanie tvarov a rozmerov objektov nachádzajúcich sa v mračne bodov snímanej scény.

V prvej časti práce sú predstavené základné metódy bezkontaktného získavania hĺbkových dát. Metódy sú následne navzájom porovnané v kapitole 1.4. Každá z metód získavania hĺbkových dát je pre niektoré aplikácie vhodnejšia než ostatné, preto je pri výbere metódy snímania hĺbkových dát nutné zvážiť podmienky danej aplikácie. V tabuľke 1.11 sú jednotlivé metódy porovnané z hľadiska podmienok použitia.

Druhou časťou práce je vytvorenie komunikačného protokolu pre uľahčenie získavania a spracovania dát zo senzorov RealSense. Výrobcom dodávaný podporný balík je pre použitie senzoru v prostredí MATLAB príliš komplikovaný. Z toho dôvodu je podporný balík spracovaný vo forme jedného objektu, ktorého metódy slúžia pre nastavenie zariadenia a získavanie všetkých typov podporovaných dát. Zariadenie je pri použití tohto objektu možné použiť nie len ako zariadenie pracujúce v skutočnom čase, ale aj ako nahrávač a prehrávač. Rýchlosť získavania snímok je dostatočná pre plynulé snímanie scény. Objekt vytvorený v tejto práci je vhodným základom pre ďalšie práce, pri ktorých bude využitý senzor RealSense.

Dáta získané z objektu popisovaného v druhej časti tejto práce je následne potrebné spracovať. Spracovaním získaných dát sa zaoberá tretia časť práce. Zo senzorov RealSense je možné získať štyri druhy dát. Prvé tri typy sú získavané vo formáte obrázku. Jedná sa o dáta získané z farebnej kamery, ktorejkoľvek z dvojice infračervených kamier a hĺbkovej snímky vytvorenej spojením dvoch infračervených snímok. Štvrtým typom dát je mračno bodov pozostávajúce z údajov o priestorových súradniciach bodu a jeho farebného zloženia. Prostredníctvom objektu RSIO, ktorý je predstavený v kapitole 4, je možné tieto typy dát spracovať a zobrazovať. RSIO je zameraný na spracovanie mračna bodov, ktoré je možné podľa potreby transformovať, orezávať na oblasť záujmu a hľadať v ňom objekty tvaru roviny, gule a valca. Pre ďalšiu prácu s dátami je následne možné previesť takto upravené mračno bodov na formát farebného obrázku, ktorý je možné ďalej spracovávať. Poslednou časťou práce je implementácia získavania a spracovania dát zo senzoru RealSense do mechatronického exponátu. Aj napriek tomu, že pôvodne návrh exponátu nemal byť súčasťou tejto práce, bolo potrebné zariadenie navrhnuť a vyrobiť, a tak je návrh a výroba exponátu zakomponovaná ako súčasť práce.

Ako mechatronický exponát je vybraný manipulátor so štyrmi stupňami voľnosti. Pre jeho riadenie je vytvorený objekt MO popísaný v kapitole 5.6. Úlohou tohto objektu je výpočet uhlov natočenia jednotlivých aktuátorov a vyhodnotenie schopnosti manipulátoru dosiahnuť objekt, ktorý má byť presunutý. Diely exponátu sú vyrobené metódou 3D tlače, čo je dôvodom nedostatočnej tuhosti a tým zapríčinennej nepresnosti pri polohovaní zariadenia. Tento problém je možné vyriešiť spevnením konštrukcie v oblasti kĺbov a použitím kovových ozubených súkolesí pre prenos krútiaceho momentu z aktuátorov. Klasifikácia objektov taktiež nie je vždy presná. Táto funkcionalita je závislá na kvalite získaného mračna bodov. Kvalitu klasifikácie je možné

## 5 NÁVRH MECHATRONICKÉHO EXPONÁTU

zlepšiť dodatočným osvetlením snímanej scény alebo zvolením inej polohy senzoru vytvárajúceho hĺbkovú mapu. Aj napriek týmto nedostatkom je tento exponát dobrým prostriedkom pre prezentáciu komunikačného protokolu pre získavanie a spracovávanie dát zo senzoru RealSense vytvoreného a popísaného v tejto práci.

# Literatúra

- [1] LI, Larry. *Time-of-Flight Camera – An Introduction*. [online]. Texas Instruments, Január 2014. Revízia Máj 2014 [cit. 2020-05-10]. Dostupné z: <http://www.ti.com/lit/wp/sloa190b/sloa190b.pdf>
- [2] DAQRI. DEPTH CAMERAS FOR MOBILE AR: FROM IPHONES TO WEARABLES AND BEYOND. In: *Medium.com*. [online]. 26.4.2020 [cit. 2020-05-10]. Dostupné z: <https://medium.com/@DAQRI/depth-cameras-for-mobile-ar-from-iphones-to-wearables-and-beyond-ea29758ec280>
- [3] 3-D Sensors Bring Depth Discernment to Embedded Vision Designs. *Edge-ai-vision.com*. [online]. 12.7.2013 [cit. 2020-05-10]. Dostupné z: <https://www.edge-ai-vision.com/2013/07/3-d-sensors-bring-depth-discernment-to-embedded-vision-designs/>
- [4] BAKUŁA, Krzysztof. *Multispectral airborne laser scanning - a new trend in the development of LiDAR technology* [online]. Varšava, 2015 [cit. 2020-05-19]. Dostupné z: [https://www.researchgate.net/publication/296486863\\_Multispectral\\_airborne\\_laser\\_scanning\\_-\\_a\\_new\\_trend\\_in\\_the\\_development\\_of\\_LiDAR\\_technology](https://www.researchgate.net/publication/296486863_Multispectral_airborne_laser_scanning_-_a_new_trend_in_the_development_of_LiDAR_technology)
- [5] Velodyne Lidar partners with Nikon to mass-produce cheaper lidar sensors. In: *The robot report.com*. [online]. Revízia 27.4.2019 [cit. 2020-05-19]. Dostupné z: <https://www.therobotreport.com/velodyne-lidar-partners-with-nikon-to-mass-produce-cheaper-lidar-sensors/>
- [6] SPERL, Peter. Microsoft Kinect Depth Sensing. In: *Anyline.com*. [online]. [cit. 2020-05-12]. Dostupné z: [https://anyline.com/wp-content/uploads/2017/10/peter\\_kinect\\_presentation.pdf](https://anyline.com/wp-content/uploads/2017/10/peter_kinect_presentation.pdf)
- [7] A Comprehensive List of 3D Sensors Commonly Leveraged in ROS Development. In: *Rosindustrial.org*. [online]. 2016 [cit. 2020-05-12]. Dostupné z: <https://rosindustrial.org/3d-camera-survey>
- [8] IFM O3D303. In: *Ifm.com*. [online]. [cit. 2020-05-12]. Dostupné z: <https://www.ifm.com/us/en/product/O3D303?tab=details>  
Path: Homepage; Products; Vision sensors; 3D cameras; 3D cameras; O3D303
- [9] GERIG, Guido. *Structured Light II* [online]. 2013 [cit. 2020-04-15]. Dostupné z: <http://www.sci.utah.edu/~gerig/CS6320-S2013/Materials/CS6320-CV-S2013-StructuredLight-II.pdf>
- [10] GENG, Jason. Structured-light 3D surface imaging: a tutorial. *Advances in Optics and Photonics* [online]. Optical Society of America, 2011, roč. 3, č. 2, s. 128-160 [cit. 2020-05-15]. DOI: 10.1364/AOP.3.000128. Dostupné z: [http://www.rtbasics.com/Downloads/IEEE\\_structured\\_light.pdf](http://www.rtbasics.com/Downloads/IEEE_structured_light.pdf)

- [11] XtionPro Live. In: *Asus.com*. [online]. [cit. 2020-05-16]. Dostupné z: [https://www.asus.com/media/global/products/hahEFPMWY9UVDL7z/ODjGCbIYqA23dyML\\_500.jpg](https://www.asus.com/media/global/products/hahEFPMWY9UVDL7z/ODjGCbIYqA23dyML_500.jpg)
- [12] XtionPro Live. In: *Asus.com*. [online]. [cit. 2020-05-16]. Dostupné z: [https://www.asus.com/3D-Sensor/Xtion\\_PRO\\_LIVE/overview/](https://www.asus.com/3D-Sensor/Xtion_PRO_LIVE/overview/)
- [13] Zivid One+. In: *Zivid.com*. [online]. [cit. 2020-05-18]. Dostupné z: <https://www.zivid.com/zivid-one-plus> Path: Menu; 3D Cameras; Zivid One Plus
- [14] Zivid One+ Technical specification. *Zivid*[online]. [cit. 2020-05-18]. Dostupné z: <https://www.zivid.com/hubfs/files/SPEC/Zivid%20One%20Plus%20Datasheet.pdf>
- [15] KOVACOVSKY, Tomas a ZIZKA, Jan. PhoXi@3D Camera. *Photoneo s.r.o* [online]. 2018 [cit. 2020-05-18]. Dostupné z: [https://www.messestuttgart.de/vision/fileadmin/media/besucher/Award/Abstracts/VISION\\_18\\_Abstract\\_Photoneo.pdf](https://www.messestuttgart.de/vision/fileadmin/media/besucher/Award/Abstracts/VISION_18_Abstract_Photoneo.pdf)
- [16] PhoXi 3D Scanner. In: *Photoneo.com*. [online]. [cit. 2020-05-18]. Dostupné z: <https://www.photoneo.com/phoxi-3d-scanner/> Path: Menu PhoXi 3D Scanner
- [17] BOUAIN, Mokhtar a ALI, Karim a BERDJAG, Denis a FAKHAFAKH, Nizar a ATITALLAH, Rabie. An Embedded Multi-Sensor Data Fusion Design for Vehicle Perception Tasks. *Journal of Communications*. ResearchGate, 2018, **13**(1), 18-14. DOI: 10.12720/jcm.13.1.8-14. Dostupné z: [https://www.researchgate.net/publication/322722003\\_An\\_Embedded\\_Multi-Sensor\\_Data\\_Fusion\\_Design\\_for\\_Vehicle\\_Perception\\_Tasks](https://www.researchgate.net/publication/322722003_An_Embedded_Multi-Sensor_Data_Fusion_Design_for_Vehicle_Perception_Tasks)
- [18] MultiSense S7. in: *Carnegierobotics.com*. [online]. [cit. 2020-05-20]. Dostupné z: <https://carnegierobotics.com/multisense-s7>
- [19] Meet ZED. In: *Stereolabs.com*. [online]. [cit. 2020-05-20]. Dostupné z: <https://www.stereolabs.com/zed/> Path: Menu; Products; ZED
- [20] Meet ZED 2. In: *Stereolabs.com*. [online]. [cit. 2020-05-20]. Dostupné z: <https://www.stereolabs.com/zed-2/> Path: Menu; Products; ZED 2
- [21] Meet ZED Mini, the world's first camera for mixed-reality. In: *Stereolabs.com*. [online]. [cit. 2020-05-20]. Dostupné z: <https://www.stereolabs.com/zed-mini/> Path: Menu; Products; ZED Mini
- [22] Depth Camera D415. In: *Intelrealsense.com*. [online]. [cit. 2020-05-20]. Dostupné z: <https://www.intelrealsense.com/depth-camera-d415/> Path: Products; Stereo Depth; Depth Camera D415
- [23] Depth Camera D435. In: *Intelrealsense.com*. [online]. [cit. 2020-05-20]. Dostupné z: <https://www.intelrealsense.com/depth-camera-d435/> Path: Products; Stereo Depth; Depth Camera D435
- [24] Depth Camera D435i. In: *Intelrealsense.com*. [online]. [cit. 2020-05-20]. Dostupné z: <https://www.intelrealsense.com/depth-camera-d435i/> Path: Products; Stereo Depth; Depth Camera D435i

- [25] *Intel RealSense D400 Series Product Family, Datasheet*. [online]. Intel, 2019, Revízia 005 [cit. 2020-04-15]. Dostupné z: <https://www.intel.com/content/dam/support/us/en/documents/emerging-technologies/intel-realsense-technology/Intel-RealSense-D400-Series-Datasheet.pdf>
- [26] GRUNNET-JEPSEN, Anders a SWEETSER N. John a WOODFILL, John. *Best-Known-Methods for Tuning Intel® RealSense™ D400 Depth Cameras for Best Performance*. [online]. Revízia 1.9 [cit. 2020-06-07]. Dostupné z: [https://www.intel.com/content/dam/support/us/en/documents/emerging-technologies/intel-realsense-technology/BKMs\\_Tuning\\_RealSense\\_D4xx\\_Cam.pdf](https://www.intel.com/content/dam/support/us/en/documents/emerging-technologies/intel-realsense-technology/BKMs_Tuning_RealSense_D4xx_Cam.pdf)
- [27] *INI Config* [online databáza]. Verzia 1.3.0.0. Mathworks, 22.3.2010 [cit. 2020-06-07]. Dostupné z: <<https://www.mathworks.com/matlabcentral/fileexchange/24992-ini-config>>. MathWorks File Exchange, Autor: Evgeny Pr
- [28] RMSE: Root Mean Square Error. In: *Statisticshowto.com*. [online]. [cit. 2020-06-07]. Dostupné z: <https://www.statisticshowto.com/probability-and-statistics/regression-analysis/rmse-root-mean-square-error/>
- [29] TORSHIN, Sergey. *Tower Pro SG90 9g* [online databáza]. 22.10.2019 [cit. 2020-06-07]. Dostupné z: <<https://grabcad.com/library/tower-pro-sg90-micro-servo-2>>. GrabCAD Community
- [30] KOBER, Filip. *Stepper Motor 28BYJ-48-5V* [online databáza]. 17.4.2015 [cit. 2020-06-07]. Dostupné z: <<https://grabcad.com/library/stepper-motor-28byj-48-5v-1>>. GrabCAD Community
- [31] Arduino Nano. In: *Store.arduino.cc*. [online]. [cit. 2020-06-07]. Dostupné z: <https://store.arduino.cc/arduino-nano>  
Path: Menu; Store; ARDUINO BOARDS & MODULES; Arduino Nano
- [32] Martin, S. Servo motor. In: *Navody.arduino-shop.cz* [online]. 10.3.2017 [cit. 2020-06-07]. Dostupné z: <https://navody.arduino-shop.cz/arduino-projekty/servo-motor.html>
- [33] KUMAR, V. *Robot Geometry and Kinematics* [online]. [cit. 2020-05-05]. Dostupné z: <https://www.seas.upenn.edu/~meam520/notes02/IntroRobotKinematics5.pdf>

# Zoznam skratiek

- CW** Continuous Wave - kontinuálna vlna
- DLP** Digital Light Processing - digitálne spracovanie svetla
- DSP** Digital Signal Processor - digitálny signálový procesor
- GUI** Graphical User Interface - grafické užívateľské prostredie
- IMU** Inertial Measurement Unit - vnútorná meracia jednotka
- IPD** Inter-Pupillary Distance - vnútro-zrenicová vzdialenosť
- IR** Infrared - infračervené
- LED** Light Emitting Diode - svetlo vyžarujúca dióda
- LIDaR** Light Detection and Ranging - detekcia a dosah svetla
  - MO** Manipulator Object - objekt slúžiaci pre komunikáciu a simuláciu polohy manipulátoru
  - PC** Point Cloud - mračno bodov, objekt pointCloud
- PWM** Pulse Width Modulation - modulácia šírky pulzov
- RMSe** Root Mean Square error - chyba kvadratického priemeru
- RSIO** RSImage Object - objekt slúžiaci pre spracovanie dát získaných použitím
- RSO** RealSense Object - objekt slúžiaci pre komunikáciu so zariadeniami RealSense
- SL** Structured Light - štruktúrované svetlo
- SLAM** Simultaneous Location and Mapping - simultánna lokalizácia a mapovanie
  - SV** Stereo Vision - stereo snímanie
- TOF** Time Of Flight - doba letu
- USB** Universal Serial Bus - univerzálna sériová komunikačná linka
- VPU** Vision Processing Unit - mikroprocesor určený pre účely počítačového videnia
- XY** rovina 2D obrazu

# Zoznam obrázkov

1.1	Meranie vzdialenosti použitím metódy TOF [2]	10
1.2	Princíp merania vzdialenosti použitím metódy TOF s pulzným zdrojom svetla [1]	11
1.3	Princíp merania vzdialenosti použitím metódy TOF CW [1]	12
1.4	Princíp merania vzdialenosti použitím multifrekvenčnej metódy TOF [1]	12
1.5	Snímka vytvorená LIDaR zariadením [5]	13
1.6	Kamera Microsoft® Kinect™ 2.0 [7]	14
1.7	Senzor IFM O3D303 [8]	15
1.8	Princíp triangulácie - metóda SL [9]	16
1.9	Druhy vzorov pre projekciu [10]	17
1.10	Kamera Microsoft® Kinect™ 1.0 (vľavo) a premietaný vzor (vpravo)[6]	18
1.11	Senzor Asus® XtrionPro™ Live [11]	19
1.12	Senzor Zivid One+ [13]	20
1.13	Skener PhoXi 3D [16]	21
1.14	(a) schematické zobrazenie stereo párovania, (b) poloha totožného bodu v snímke pravej a ľavej kamery [17]	22
1.15	Metóda pasívneho stereo snímania (vľavo) a aktívneho stereo snímania (vpravo) [2]	23
1.16	Senzor MultiSense S7 [18]	24
1.17	Senzory ZED (hore), ZED 2 (v strede) a ZED Mini (dole) [19][20][21]	25
1.18	Senzory RealSense™ D415 (hore), D435 a D435i (dole) [22][23]	26
1.19	Znázornenie efektu zatienenia [2]	27
2.1	Infračervená snímka z modelu D415 s viditeľným premietaným vzorom	29
3.1	Diagram popisujúci činnosť konštruktéru	33
3.2	GUI slúžiace na nastavenie rozlíšenia	34
3.3	Diagram popisujúci princíp nastavenia zariadenia pred spustením	35



## ZOZNAM OBRÁZKOV

3.4	Diagram popisujúci princíp metódy <i>start()</i> (spustenie dátového toku) . . . . .	36
3.5	Všetky dostupné nastavenia a informácie o nich hĺbkového senzoru modelu D415 . . . . .	37
3.6	Snímky z GUI slúžiaceho na zmenu nastavení zariadenia RealSense . . . . .	38
3.7	Diagram popisujúci princíp získavania dát zo zariadenia RealSense . . . . .	40
3.8	Diagram popisujúci princíp získavania dát pre mračno bodov . . . . .	41
3.9	Diagram znázorňujúci proces filtrovania hĺbkovej snímky . . . . .	43
3.10	Diagram popisujúci princíp transformácie dát . . . . .	44
3.11	Graf zobrazujúci rýchlosť získavania, spracovania a vykreslenia farebných dát . . . . .	46
3.12	Graf zobrazujúci rýchlosť získavania, spracovania a vykreslenia hĺbkových dát . . . . .	46
3.13	Graf zobrazujúci rýchlosť získavania, spracovania a vykreslenia infračervených dát . . . . .	47
3.14	Graf zobrazujúci rýchlosť získavania, spracovania a vykreslenia farebných a hĺbkových dát . . . . .	47
3.15	Graf zobrazujúci rýchlosť získavania, spracovania a vykreslenia mračna bodov . . . . .	48
3.16	Graf závislosti chyby merania na vzdialenosti objektu od kamery . . . . .	49
3.17	Snímka zo zariadenia RealSense D415 vytvorená pri testovaní presnosti merania . . . . .	50
4.1	Transformácia PC metódou <i>transform_pc()</i> . Vľavo pôvodný PC, vpravo transformovaný PC . . . . .	52
4.2	Odstránenie bodov mimo oblasť záujmu. Vľavo pôvodný PC, vpravo PC redukovaný na oblasť záujmu . . . . .	52
4.3	Vyhľadávanie tvarov v PC. Vľavo pôvodný PC, vpravo PC zostavený z bodov, ktoré boli označené ako súčasť nájdeného tvaru . . . . .	53
4.4	Segmentácia PC na samostatné objekty. Vľavo zosnímaný obraz, vpravo obraz vytvorený po segmentovaní objektov podľa vzdialenosti . . . . .	54
5.1	Model prvej verzie manipulátoru . . . . .	55
5.2	Určovanie počiatočnej pozície využitím formy spínača . . . . .	56
5.3	Schéma obvodu využívaného pri určovaní základnej polohy krokového motora . . . . .	56
5.4	Mechanizmus čelustí prvej verzie manipulátoru . . . . .	57
5.5	Model druhej verzie manipulátoru . . . . .	58
5.6	Kuželové ozubené súkolesie použité pre prenos krútiaceho momentu . . . . .	58
5.7	Mechanizmus prevodu rotačného pohybu na translačný v čelustiach modelu 2.0 . . . . .	59
5.8	Manipulátor postavený na podložke s držiakom kamery, telieskami a ich zásobníkmi . . . . .	60
5.9	Schéma elektrického zapojenia manipulátoru . . . . .	61


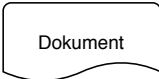
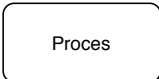
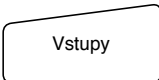
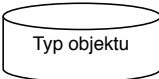
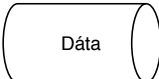

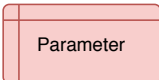
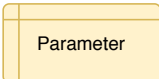
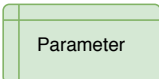

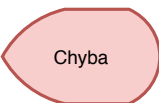
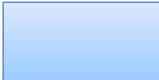
## ZOZNAM OBRÁZKOV

5.10	Náčrt manipulátoru s označenými dĺžkami ramien a uhlami; a) v karteziánskom súradnicovom systéme, b) priemet do roviny DZ, c) náčrt pre výpočet pomocou kosínovej vety . . . . .	62
5.11	Diagram popisujúci priebeh kontroly pred pohybom manipulátoru . . . . .	65
5.12	Simulácia polohy manipulátoru pre dosiahnutie objektu . . . . .	66
5.13	Diagram priebehu riadiaceho programu exponátu . . . . .	67
5.14	Zosnímanie transformovanie mračna bodov . . . . .	68
5.15	Odstránenie bodov mimo oblasť záujmu, segmentácia a klasifikácia objektov v snímanej scéne . . . . .	68
5.16	Kontrola kolízie (vľavo) a simulácia dosiahnutia objektu (vpravo) . . . . .	69
5.17	Manipulátor zdvíhajúci objekt po úspešnom ukončení simulácie . . . . .	69
A1	Legenda pre prácu s diagrammi . . . . .	80

# Zoznam tabuliek

1.1	Základné parametre senzoru Microsoft® Kinect™ 2.0 [6]	14
1.2	Základné parametre senzoru IFM O3D303 [8]	15
1.3	Základné parametre senzoru Microsoft® Kinect™ 1.0 [6]	19
1.4	Základné parametre senzoru Asus® XtrionPro™ Live [7]	19
1.5	Základné parametre senzoru Zivid One <sup>+</sup> [14]	19
1.6	Rozdiely v parametroch senzorov Zivid One <sup>+</sup> S, M a L [14]	20
1.7	Základné parametre skenerov PhoXi 3D [16]	21
1.8	Základné parametre senzoru MultiSense S7 [7]	24
1.9	Parametre senzorov ZED, ZED 2 a ZED Mini [19][20][21]	25
1.10	Základné parametre senzorov Intel® RealSense™ D415, D435 a D435i [22][23][24]	26
1.11	Porovnanie metód snímania hĺbkovej mapy [1][2][3][7]	28

# Prílohy

	Začiatok alebo koniec diagramu
	Dáta vyčítané zo súboru
	Funkcia alebo činnosť
	Vstupy do funkcií zadávané užívateľom
	Označenie vytvorenia objektu daného typu
	Výstupné dáta
	Rozhodovacia funkcia (if/else alebo switch/case)
	Vyčítanie alebo zapísanie parametru zo skupiny chránených parametrov, neprístupný užívateľovi
	Vyčítanie alebo zapísanie parametru zo skupiny nemenných parametrov, užívateľovi prístupný iba na čítanie
	Vyčítanie alebo zapísanie parametru zo skupiny verejných parametrov, prístupný na čítanie aj zápis
	Vytvorí varovanie, procesy pokračujú ďalej
	Vznikla chyba zapríčiňujúca predčasné ukončenie procesu
	Funkcia, definovaná viacerými bločkami

Obr. A1: Legenda pre prácu s diagrammi